

# 人の生活環境における自律ナビゲーションを目的とした移動ロボットの自己位置推定

著者	山田 大地
発行年	2014
学位授与大学	筑波大学 (University of Tsukuba)
学位授与年度	2013
報告番号	12102甲第6867号
URL	<a href="http://hdl.handle.net/2241/00123458">http://hdl.handle.net/2241/00123458</a>

人の生活環境における  
自律ナビゲーションを目的とした  
移動ロボットの自己位置推定

山田 大地

システム情報工学研究科  
筑波大学

2014年 3月

# 目次

<b>第1章</b>	<b>序論</b>	<b>1</b>
1.1	研究の背景	3
1.2	研究の目的	4
1.3	本論文の構成	5
<b>第2章</b>	<b>人の生活環境における自己位置推定の課題</b>	<b>7</b>
2.1	自己位置推定	7
2.1.1	自己位置推定と地図	9
2.1.2	デッドレコニング	13
2.1.3	地図上の絶対位置推定	16
2.1.4	自己位置の融合	25
2.2	生活環境における自己位置推定の課題	29
2.2.1	地図上の絶対位置推定における誤推定への対策	29
2.2.2	ランドマークの不足への対策	31
2.2.3	地図上の絶対位置推定の検討	32
2.2.4	デッドレコニングの検討	37
<b>第3章</b>	<b>高所特徴のスキャンマッチングによる自己位置推定</b>	<b>42</b>
3.1	関連研究	43
3.2	スキャンマッチングとその生活環境における課題	44
3.2.1	ICP アルゴリズム	44
3.2.2	環境の変化による影響	47
3.3	高所の計測方法	48
3.4	実験	49

3.4.1	ハードウェア	50
3.4.2	高所特徴によるスキャンマッチングを用いた自己位置推定	51
3.5	動作計画	53
3.5.1	走行結果	54
3.6	成果	56
<b>第4章</b>	<b>統計量に基づくレーザスキャナの尤度場を用いた自己位置推定</b>	<b>59</b>
4.1	関連研究	60
4.2	生活環境における不安定なスキャンデータ	62
4.3	統計量に基づくレーザスキャナの尤度場	63
4.3.1	尤度の計算	65
4.3.2	地図とスキャンデータとの一致度評価	70
4.3.3	地図とスキャンデータの一致度評価の比較	71
4.4	自律走行実験	77
4.4.1	ハードウェア	78
4.4.2	自己位置推定	78
4.4.3	走行結果	79
4.5	検討	80
4.6	成果	82
<b>第5章</b>	<b>オドメトリの系統誤差の軽減</b>	<b>84</b>
5.1	オドメトリとその誤差	85
5.1.1	オドメトリの利用	85
5.1.2	オドメトリの誤差	86
5.2	オドメトリの修正手法	88
5.2.1	オドメトリ誤差地図の作成	89
5.2.2	オドメトリの修正	91
5.3	評価実験	92
5.3.1	舗装路面の横断勾配	94
5.3.2	タイルカーペット	97



5.4 検討 . . . . .	99
5.4.1 オドメトリの誤差計測の間隔 . . . . .	100
5.4.2 セルのサイズ . . . . .	100
5.4.3 オドメトリ以外の自己位置推定が困難な環境 . . . . .	101
5.5 成果 . . . . .	102
<b>第 6 章 結論</b>	<b>105</b>
<b>謝辞</b>	<b>108</b>
<b>参考文献</b>	<b>108</b>

## 表 目 次

5.1	横断勾配往路における推定誤差の平均値 . . . . .	96
5.2	横断勾配復路における推定誤差の平均値 . . . . .	98
5.3	横断勾配復路における推定誤差の平均値 . . . . .	99
5.4	スキャンマッチングが困難な環境における静止位置 . . . . .	102

# 目 次

2.1	地図の座標系とロボットの自己位置 . . . . .	8
2.2	自己位置の追跡のシステム構成 . . . . .	9
2.3	地図の歪み . . . . .	10
2.4	地図の歪みによる自己位置推定の問題 . . . . .	11
2.5	デッドレコニングによる地図上の自己位置の推定 . . . . .	13
2.6	デッドレコニングの誤差の累積 . . . . .	14
2.7	地図上の絶対位置推定 . . . . .	17
2.8	占有格子地図の例 . . . . .	18
2.9	明示的な対応付け . . . . .	19
2.10	暗黙の対応付け . . . . .	20
2.11	誤推定による自己位置推定の破綻 . . . . .	21
2.12	偽のランドマークに対する明示的な対応付け . . . . .	22
2.13	偽のランドマークに対する暗黙の対応付け . . . . .	23
2.14	拡張カルマンフィルタによる自己位置推定 . . . . .	26
2.15	スキャンデータ . . . . .	33
2.16	独立二輪駆動型ロボットのキネマティクスパラメータ . . . . .	39
2.17	オドメトリによる自己位置推定 . . . . .	39
2.18	オドメトリの誤差と正規分布 . . . . .	41
3.1	スキャンマッチングにおける座標系 . . . . .	44
3.2	ICP アルゴリズム . . . . .	46
3.3	環境の変化によるランドマークの観測の妨げ . . . . .	47
3.4	環境の変化と地図の誤対応付け . . . . .	48
3.5	2次元レーザスキャナによる高所の計測方法 . . . . .	48

3.6	実験環境	50
3.7	ロボットの外観	51
3.8	上部レーザスキャナの取り付け	51
3.9	高所特徴によるスキャンマッチングを用いた自己位置推定	52
3.10	高さによるスキャンデータの違い	53
3.11	経路地図	54
3.12	障害物回避	55
3.13	つくばチャレンジ 2010 における高所特徴によるスキャンマッチングに成功した箇所	56
3.14	高所特徴を用いたスキャンマッチングに失敗した区間	57
4.1	生活環境における不安定なスキャンデータ	62
4.2	尤度場とスキャンデータの一致度の評価	64
4.3	格子状の地図とそのセル	65
4.4	統計量に基づく反射点観測の尤度	66
4.5	4 種類のセルの分割	67
4.6	提案手法による尤度場：歩行者のいる廊下環境における尤度場	68
4.7	提案手法に尤度場：ロボットの振動により地面のスキャンデータが不安定となる環境	69
4.8	提案手法による尤度場：ロボットの振動により地面のスキャンデータが不安定となる環境における尤度場	70
4.9	地面のスキャンデータが不安定となる環境	73
4.10	一致度の評価値の比較:地面のスキャンデータが不安定となる環境	74
4.11	人通りの多い環境	75
4.12	一致度の評価値の比較:人通りの多いにおける一致度の評価	76
4.13	実験環境:つくばチャレンジ 2011 の課題コース	77
4.14	ロボットの外観	78
4.15	尤度場とスキャンデータの一致度評価を用いた自己位置推定	80
4.16	人混みを走行する様子	81

4.17 細いスロープを走行する様子 . . . . .	81
5.1 自己位置推定におけるオドメトリの役割 . . . . .	87
5.2 オドメトリ誤差地図 . . . . .	89
5.3 ロボットを初期位置に設置する方法 . . . . .	91
5.4 ロボットの外観 . . . . .	92
5.5 実験環境:横断勾配 . . . . .	93
5.6 実験環境:タイルカーペット . . . . .	93
5.7 横断勾配往路におけるオドメトリ誤差地図 . . . . .	95
5.8 横断勾配往路における走行軌跡 . . . . .	96
5.9 横断勾配往路における静止位置 . . . . .	96
5.10 横断勾配復路におけるオドメトリ誤差地図 . . . . .	97
5.11 横断勾配復路における走行軌跡 . . . . .	97
5.12 横断勾配復路における静止位置 . . . . .	97
5.13 タイルカーペットにおけるオドメトリ誤差地図 . . . . .	99
5.14 タイルカーペットにおける走行軌跡 . . . . .	99
5.15 タイルカーペットにおける静止位置 . . . . .	99
5.16 スキャンマッチングが困難な環境 . . . . .	102
5.17 スキャンマッチングが困難な環境におけるオドメトリ誤差地図 . . . . .	103
5.18 スキャンマッチングが困難な環境における走行軌跡 . . . . .	103

# 第1章 序論

2005 年の愛・地球博における 2020 年の実用化を目指したプロトタイプロボット [1] のデモ展示において，様々なロボットが披露された．この中には，人の生活支援を目的として，ユーザに飲み物や新聞を届けるロボットや高齢者や幼児の見守りのために人を追従するロボットなど人と共生するロボットが紹介されている．このような，人の身近で支援するロボットは高齢化社会における労働力や要介護者の生活支援などのサービス，また産業としての期待が高まっている．人の身近で支援するロボットには，人と同じ環境で動作することが求められる．ロボットの動作のうち移動の動作が実現すると，ロボットの活動範囲の拡張，人に随伴するサービス，能動的なサービスが可能となる．ロボットが移動するための技術として，目的地まで自律的に移動する自律ナビゲーションがあり，研究，開発が盛んに行われている．自律ナビゲーションが実現されると，道案内や荷物の搬送，巡回警備などのサービスを提供するロボットの開発に役立つ．このため，自律ナビゲーションの研究が盛んに行われている．その例として，自動運転車がある．自動運転車は 2010 年に Google が開発中であることを発表し，自動車メーカー各社も相次いで自動運転車の開発に対する公表をした．2013 年 11 月 9 日には国会議事堂周辺の公道で自動運転車の実証実験にトヨタ自動車，日産自動車，ホンダが参加し，各社の自動運転車に安倍晋三首相が試乗した [2]．また，歩道や屋内環境などで活動するロボットについての研究も盛んである．2007 年より毎年，実際の歩道 1km 強のコースを走行する自律移動ロボットのチャレンジであるつくばチャレンジ [3] が開催されている．2013 年には研究機関を中心に 44 チームが参加し，実際の生活環境で人が活動する中動作する自律移動ロボットの研究が盛んに行われている．しかし，生活環境における自律ナビゲーションは難しく，つくばチャレンジ 2013 において課題を達

成したロボットは3台にとどまっている[4]。

ロボットが自律的に動作するには、目的を達成するための動作を計画し、その動作を実行する。この動作を計画の際には、周囲の状況や目的地の方向や距離に応じて動作を決定する。そこで、ロボットは周囲の状況を知るために、センサより得られた情報をもとに環境を認識する必要がある。しかし、人の生活環境では環境認識が困難となる。従来、ロボットは環境の認識が容易な工場などで活動することが多かった。工場では、ロボットのために環境の認識が容易となるように整備できる。これに対して、生活環境は人のために整備されており、景観やコストなどの理由からロボットのために環境を整備できるとは限らない。工場で稼働するロボットは、特定の工場内のみで活動できればよいが、人の身近で支援するロボットは、人と同等の範囲を活動することが望ましい。人の活動範囲において、その環境は屋外、屋内、オフィス、家庭、市街地、公園など多様であり、この環境の多様性も環境の認識が困難な原因である。

自律ナビゲーションでは、目的地に到達するための動作を計画し、その動作を実行する。このため、周囲の環境より「どこにいるのか?」、すなわち現在地から目的地までの経路や方向、距離を認識する必要がある。本研究では、自律ナビゲーションにおける環境の認識のうち「どこにいるのか?」を認識するために、自己位置推定について研究した。自己位置推定では、センサの計測より地図上のロボットの位置、姿勢(自己位置)を推定する。地図とはロボットの走行する環境の情報を記述したものであり、この地図上で目的地を指定し、自己位置を推定することで、目的地までの経路や方向、距離を知ることができる。

本研究では、生活環境における自己位置推定の問題を詳細に考察し、生活環境においてロバストな自己位置推定手法を提案する。以降、本章では人の生活環境における自律ナビゲーションについて問題を整理し、本研究の目的について述べる。

## 1.1 研究の背景

近年，人の身近で生活を支える目的で，人と共生するロボットが研究されている [5][6][7][8][9][10][11]．このような人と共生するロボットは，人と同じ環境で動作することが求められる．ロボットが人と同じ環境で，目的地までの自律的に移動する動作を可能となると道案内，警備，配達などの作業が可能となる．目的地までの自律的に移動するロボットの機能を自律ナビゲーションと呼び，自律ナビゲーションを用いたサービスを提供するロボットについて研究されている [5][6][7][8][9]．しかし，これらのロボットは環境中にロボットの活動のための目印となる磁気テープや反射テープを設置したり，デモや実験のために人の出入りを制限した環境での動作であり，あるがままの環境で自律ナビゲーションを実現するには至っていない．

自律的に動作するロボットは，周囲と自身の状況に応じて動作を決定する．このため，ロボットは周囲の状況を知るために，センサより得られた情報をもとに環境を認識する必要がある．従来よりロボットが活躍していた工場などでは，環境の認識を容易とするため，磁気テープは反射材などのセンサの計測から特定が容易な目印を設置するなど，ロボットのために環境を整備していた．また，人の出入りを制限することで環境の変化を少なくし，認識を容易としている．しかし，人の生活環境では景観や工事のコストなどの理由から，磁気テープなどの目印を設置できないことがある．同時に人の活動により環境が激しく変化することがあるため，環境の認識は困難となる．さらに，生活環境は歩行者の移動，季節，時刻による変化や，日照やロボットの移動による振動のため，センサの計測はバラつき，不安定となる．また，人の活動範囲には屋外，屋内，オフィス，家庭，市街地，公園など多様な環境があり，これらの環境でロボストに機能することが求められる．

自律ナビゲーションでは，目的地までの動作を計画し，その動作を実行する．このため，周囲の環境より「どこにいるのか?」を認識する必要があり，自己位置推定が用いられる．自己位置推定では，センサの計測より地図上のロボットの位置，姿勢(自己位置)を推定する．この時，地図上で目的地の位置を指定し



ておくことで，目的地までの方向や距離を知ることができる．自己位置推定では，ロボットの制御動作から推定したり，センサの計測から得られる周囲の環境の情報をもとに推定する．しかし，生活環境のように整備されていない環境では，制御した通りに動作せず，誤差が生じることがある．また，周囲の環境が変化し，さらにロボットの振動などによりセンサの計測が不安定であるため，自己位置推定に有意な環境の情報をえることは困難となる．

## 1.2 研究の目的

本研究では，人の生活環境における自己位置推定の課題について考察し，市街地，公園，オフィス環境等の多様な生活環境において有効に働く自己位置推定手法を提案する．これにより，人の身近な環境で，ロボットが自在に移動可能となることを目的としている．ここで，本研究では人の生活環境として，ロボットのために環境を操作しない，また，人の出入りを制限しない，あるがままの環境を対象に研究を行った．また，長距離の走行が可能である車輪型移動ロボットの自己位置推定について取り組んだ．このため，対象とする環境は車輪による走行が可能な環境に限られるが，一般的に人が生活する環境では路面が整備されているため，広い範囲で適用可能であると考えられる．

自己位置推定は大別して，ロボットの動作から位置，姿勢を推定するデッドレコニングと，周囲の環境の認識をもとに地図上の絶対位置を推定する方法がある．デッドレコニングはセンサの計測から推定されるロボットの動作をもとに，既知の自己位置からの相対的な自己位置を推定する方法である．しかし，デッドレコニングでは推定したロボットの動作には誤差が含まれるため，ロボットが移動するにつれて自己位置の推定誤差が増大する．そこで，地図上の絶対位置推定により誤差を修正する．地図上の絶対位置推定では，地図の情報と自律走行時の計測の情報が一致する位置，姿勢を推定するため，誤差は累積しない．しかし，地図上の絶対位置推定は一般的に計算負荷が大きく，間欠的にしか自己位置を推定できない．また，多くの地図上の絶対位置推定の手法は，計算を容易とするため，直前の自己位置の誤差が小さいと仮定している．このた

め、自己位置推定では、地図上の絶対位置推定とデッドレコニングを並用する。本研究では、地図上の絶対位置推定とデッドレコニングについて、それぞれの誤差と推定の誤り、その原因について考察し、解決方法を提案する。

地図上の絶対位置推定はセンサから得られる周囲の環境の情報から、地図上の位置、姿勢を推定する。このため、環境への依存が大きく、また様々なセンサを用いた手法が研究されている。生活環境においては、歩行者のなどの時間による変化や、日照やロボットの振動などの計測の条件の変化のため、センサの計測は不安定となる問題がある。地図上の絶対位置推定では、センサの計測が不安定な環境では、推定に誤りが生じることがある。そこで、この推定の誤りの原因と、各手法に対して生活環境における問題について考察する。この考察をもとに、本研究では、生活環境において不安定な計測が少ない高所の特徴を用いた自己位置推定と、複数回の計測から得られる統計量より計算される尤度を用いて自己位置を推定する手法を提案する。

デッドレコニングにおいてはその誤差について考察し、事前に走行した際のセンサデータより誤差を軽減する方法を提案する。デッドレコニングの誤差を軽減することで、長く走行しても誤差が増大しにくくなる。また、地図上の絶対位置推定が容易となり、推定の誤りが少なくなることが期待できる。

### 1.3 本論文の構成

本章では、人の生活環境における自律ナビゲーションについて問題を整理し、本研究が生活環境における自己位置推定の実現を目的としていることを述べた。次に2章では、一般的な自己位置推定の手法についてまとめ、本研究がとるアプローチの検討とその概要について述べる。次に、絶対位置推定に対する2つアプローチについて、それぞれ3章では高所の特徴を用いた自己位置推定の手法の説明と、4章では統計量に基づくレーザスキャナの尤度場を用いた自己位置推定についてその方法と実験により実際の生活環境における有効性を検証した結果を述べる。5章では、デッドレコニングの改善として、車輪型移動ロボットのデッドレコニングであるオドメトリの系統誤差を軽減する方法と、従来の

オドメトリと比較した結果について述べる．最後に 6 章にて，本研究の成果についてまとめる．

## 第2章 人の生活環境における自己位置推定の課題

本章では，移動ロボットの自己位置推定の概要と生活環境における自己位置推定の問題について整理し，本研究が取るアプローチについて検討する．初めに，自己位置推定とその生活環境における問題について述べる．次に，本研究がとるアプローチの検討とそのアプローチについて述べる．

### 2.1 自己位置推定

自律ナビゲーションはロボットが目的地まで自律的に移動する機能である．自律ナビゲーションでは，ロボットは現在の位置から目的地に移動するための動作を計画し，計画した動作を実行する．この時，目的地までの方向や距離，つまり，目的地に対する現在のロボット自身の位置を知る必要がある．自己位置推定では，地図上のロボットの位置，姿勢(自己位置)を推定する．ここで，目的地を地図上の位置で指定すると，自己位置から目的地までの経路や方向，距離を知ることができる．センサによりロボットの位置，姿勢を直接計測することはできないため，ロボットはセンサの計測から自己位置を推定する．

車輪型移動ロボットは車輪で走行可能な面上を移動するため，一般的に地図は実空間を平面に近似した2次元のメトリックな空間上で表現される．この時，図2.1に示すように，自己位置は地図の2次元平面上での位置，姿勢(向き) $\mathbf{x} = (x, y, \theta)^T$ の3自由度となる．これにより，6自由度(縦，横，高さ，ロール角，ピッチ角，ヨー角)である実空間の自己位置を推定に対して，計算負荷を減らし，また，推定を容易とすることができる．本研究では，この2次元平面上の自己位置推定を対象とした．

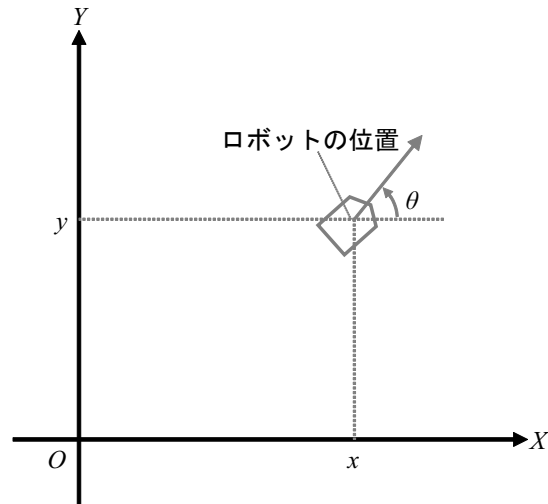


図 2.1: 地図の座標系とロボットの自己位置

自己位置推定は、直前の自己位置が既知か未知かで、大きく2つに分けられる。本論文では、直前の自己位置が既知である場合の自己位置推定を自己位置の追跡、未知である場合の自己位置推定を大域的自己位置推定と呼ぶ。大域的自己位置推定は、ロボットの起動位置が不明であったり、衝突などによりロボットが自己位置を見失うなど、特殊な状況により自己位置が未知となった場合に用いられる。本研究では、このような特殊な状況は考えないこととし、自己位置の追跡について考えることにした。自己位置の追跡では、既知の自己位置からの移動量をもとに自己位置を推定したり、現在の自己位置は直前の自己位置の付近であるとして推定を容易にすることができる。

一般的な自己位置の追跡のシステム構成図を図 2.2 に示す。自己位置推定の方法は2種類の方法がある。1つは内界センサの計測から得られるロボットの動作をもとに、自己位置を推定するデッドレコニングである。もう1つは、外界センサにより計測された情報と地図の情報が一致する位置を探索する地図上の絶対位置推定である。また、複数の異なる推定結果を統合してより正確に自己位置を推定するための手法として、自己位置の融合がある。ここで、自律ナビゲーションの自己位置推定は地図上の自己位置を推定する課題であり、地図を参照して、地図上の絶対位置を推定することは不可欠な要素となる。同時に、ここで用いる地図もまた不可欠な要素となる。

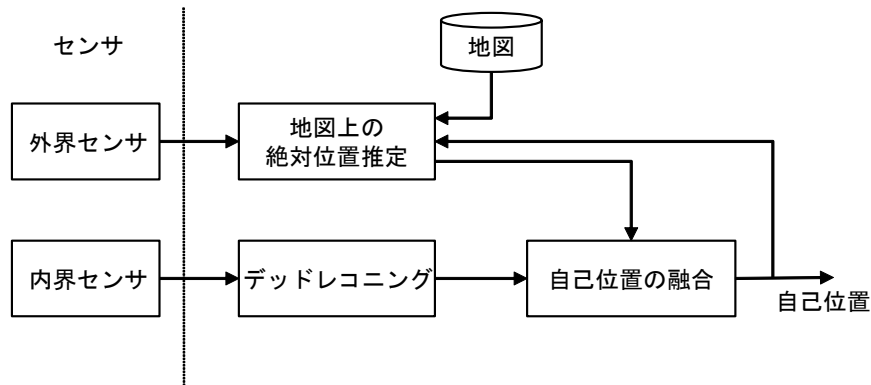


図 2.2: 自己位置の追跡のシステム構成

以降、本節では、まず自己位置推定における地図について述べる。次にデッドレコニング、地図上の絶対位置推定、自己位置の融合について、その方法と生活環境における問題を述べる。ここで、理由は2.1.3節にて述べるが、自己位置の追跡では、自己位置の誤差の増大を防ぐことが重要となるため、本節で述べる問題は、特に自己位置の推定誤差に着目している。

### 2.1.1 自己位置推定と地図

移動ロボットの自律ナビゲーションにおいて地図は、ロボットが活動する範囲の空間を表したものである。地図には、その空間における環境の情報やロボットが走行する経路を記したものがある。自律ナビゲーションにおける自己位置推定では、この地図の空間上のロボット位置、姿勢を推定する。また、自己位置推定に用いる地図には、環境の情報としてロボットの位置、姿勢が推定可能な特徴(ランドマーク)を記述する。

地図の空間は、測量によって得られた各所の位置によって定められる。この測量の誤差により、地図の空間には実空間に対して歪みが生じる。この歪みはデッドレコニングによる地図上の自己位置の推定誤差と関係する。ここで、関係すると曖昧に表現するのは、単純に地図の歪みが大きくなると誤差が増大するわけではないからである。この歪みとデッドレコニングの誤差についての詳細は2.1.2節にて、デッドレコニングの説明とともに後述することにして、ここでは地図の作成方法と地図の歪みについて述べる。

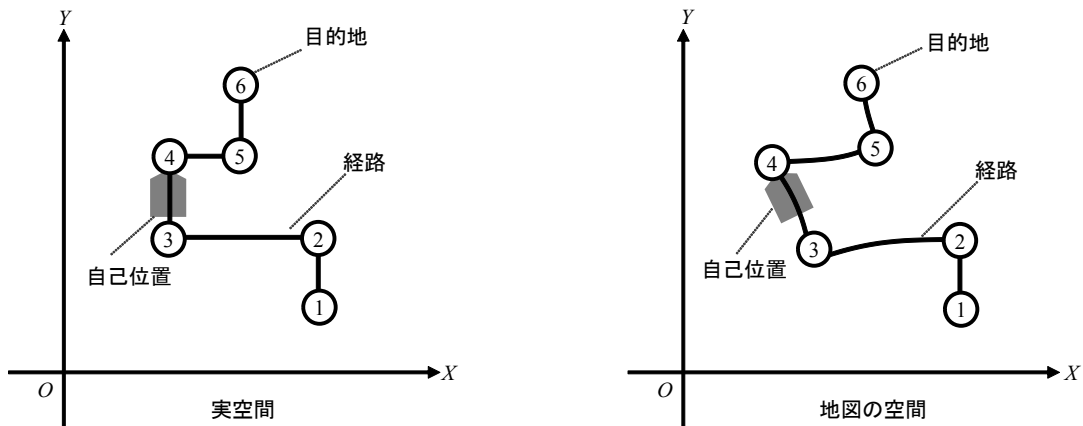


図 2.3: 地図の歪み

### 地図の作成

自己位置推定では、環境の情報としてランドマークが記された地図が必要となる。ロボットにとって環境の情報は、基本的にその環境で得られるセンサの計測であり、地図に記述された環境の情報から、環境で得られるセンサの計測もしくはその特徴と、その位置が得られることが求められる。例えば、自己位置推定では、地図に記述された静止物の位置と自己位置の推定値より測距センサの計測値を予測する。この予測と実際にセンサから得られた計測と比較して自己位置の推定値が正しいか評価する、また、別の例では、地図にはある位置、姿勢で得られたセンサの計測を記述し、自律走行時の計測と照合することで自己位置を推定する。

環境の情報を記述した地図として、間取り図などの人間の地図があるが、ロボットが走行する環境において常に地図が用意されていることは期待できない。このため、一般的に地図はロボットを操作するなどして、実際に環境中を走行した際のセンサの計測をもとに作成される。この時、地図の作成のための測量は、ランドマークの位置や走行時のロボットの位置、姿勢を定めることで実現される。また、ランドマークの位置は、センサの計測とその時のロボットの位置、姿勢により求められる。つまり、測量の誤差による地図の空間の歪みは、端的には地図の作成時の自己位置の推定誤差に起因しているといえる。また、地図の作成は、自己位置推定を内包している。図 2.3 に実空間に対する地図の空

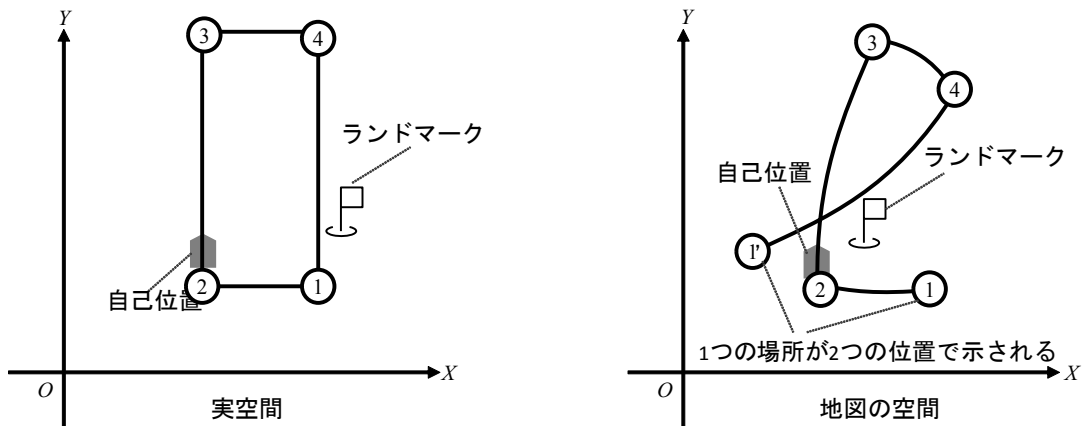


図 2.4: 地図の歪みによる自己位置推定の問題

間の歪みについて示す。図 2.3 では、実空間における各番号の場所に対して、地図の空間上における位置は、地図作成時の自己位置推定の結果によって決まり、対応する番号の位置となる。この時、地図の空間上の各場所の位置は、自己位置の推定誤差により、実空間とずれた場所になる。

地図の作成と自律ナビゲーションの自己位置推定では、技術的には共通する部分が多いが、本質的に異なる点がある。まず、地図の作成における自己位置推定は実空間におけるロボットの位置、姿勢を推定することに対し、自律ナビゲーションにおける自己位置推定は、地図の空間上におけるロボットの位置、姿勢を推定する。また、自律ナビゲーションにおける自己位置推定では、リアルタイム性が求められるが、地図の作成における自己位置推定では、リアルタイム性は必要ない。生活環境における地図の作成における自己位置推定は、自律ナビゲーションにおける自己位置推定と同様に困難ではあるが、地図の作成は失敗が許されることから、トライアンドエラーを繰り返し作成することができる。生活環境における地図の作成の例としては、人手で編集したり、GPS(global positioning system)<sup>1</sup>の計測を用いて作成している [12][13][14]。しかし、人手の作業は労力がかかり、また環境とロボットの両方に精通した人でないと編集が困難である。また、2.2.3 節にて後述するが、GPS は計測できる環境が限られるため、常に歪みの少ない地図を用意することは困難である。

<sup>1</sup>正式には衛星測位システム (GNSS: global navigation satellite system) であるが、GPS の名称で認知されることが多いため、本論文では GPS と示す



基本的に地図の空間の歪みは小さいことが好ましいが、地図の空間の歪みは致命的な問題ではない。地図の歪みによる自己位置推定の問題について図2.4に示す。図2.4に示す例では、本来、1番と4番の経路沿いにあるランドマークが、地図上では2番と3番との経路沿いにあるようにも見える。このように、ランドマークの位置が曖昧となると、このランドマークがもとで自己位置推定に誤りが生じることがあるため、問題となる。また、図2.4では、本来同じ場所である地図上の1番と1'番が別の位置で示されているように、ループする経路では同じ場所であることが認識できないことがある。このような問題があるため、基本的に地図の空間の歪みは小さいことが好ましい。しかし、ランドマークの位置が曖昧になる、地図の歪みに起因するデッドレコニングの誤差が大きいなどの問題がなく、局所的な範囲で歪みが十分小さければ、後述する地図上の絶対位置推定により地図上の自己位置を推定することができる。地図上の自己位置が推定できれば、地図が歪んでいても実空間上で対応した場所にいることが認識できる。例えば、図2.3において、地図上の自己位置が4番の位置であることが分かれば、実空間においても4番にいることが認識できるため、目的地までの動作の計画は可能となる。このことから、簡易な自己位置推定により地図を作成することがある。直前のセンサの計測を地図に見立てて自己位置を推定したり[15]、デッドレコニングのみにより自己位置を推定する[16][17]、などの方法で作成した事例がある。これらの方法は、2.1.2節にて後述するデッドレコニングと同様に誤差が累積するため、走行距離が長くなると次第に地図の空間は歪む。このため、ループや往復する区間のある環境への適用には工夫が必要となるが、このような簡易な自己位置推定方法で作成した歪んだ地図でも、自律ナビゲーションが可能な地図を作成できる。

本研究は自律ナビゲーションを目的としているため、地図が既知であるとして地図の作成における自己位置推定は研究の対象外とし、地図上の自己位置推定のみに取り組む。また、本論文ではとくに修飾がなく「自己位置」や「自己位置推定」と称する場合、自律ナビゲーションにおける自己位置およびその推定を意味し、地図の空間上におけるロボットの位置、姿勢とその推定のこととする。

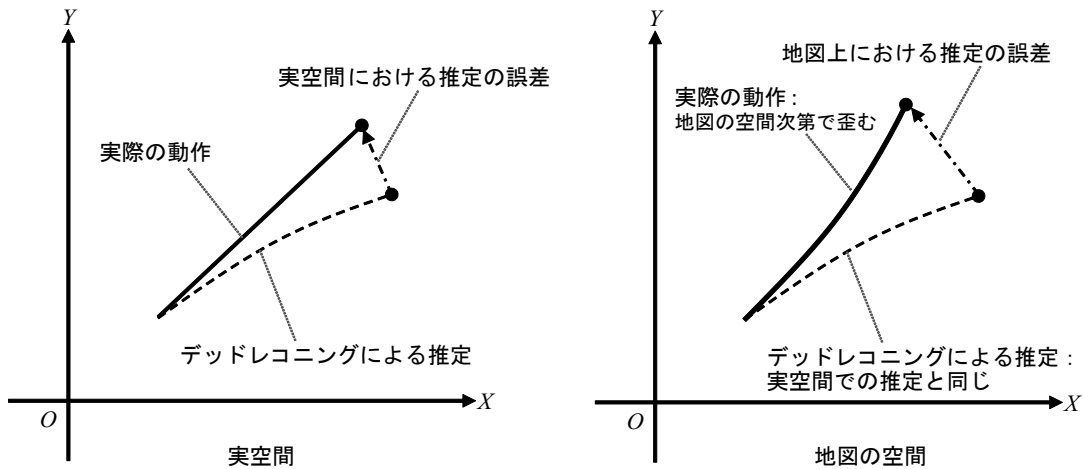


図 2.5: デッドレコニングによる地図上の自己位置の推定

### 2.1.2 デッドレコニング

移動ロボットのデッドレコニングは内界センサの計測をもとにロボットの速度、角速度または移動量を推定し、既知の初期位置にこれを累積して自己位置を推定する方法である。つまり、デッドレコニングでは、内界センサの計測よりロボットの動作を推定し、この動作をもとに既知の自己位置からの相対的な自己位置を計算する。2次元平面上の自己位置推定では、一般的に水平な平面上を移動していると仮定して、ロボットの動作を推定する。ここで、デッドレコニングは実空間上での自己位置を推定する方法であり、地図作成における自己位置推定にも用いられる。図 2.5 に示すように実空間上での動作を内界センサの計測をもとに推定する。自律ナビゲーションにおける自己位置推定に適用する際には、地図の歪みは十分小さい(少なくとも局所的には)として、実空間上での自己位置の推定値をそのまま地図上の自己位置の推定値とする。センサには車輪に搭載したエンコーダ、加速度センサ、ジャイロセンサなどが用いられる。一般的にこれらのセンサのサンプリング周期は数～数十 ms と短く、またデッドレコニングの計算負荷は小さいため、短い周期で自己位置を推定できる利点がある。

センサの計測より推定した速度、角速度などには誤差が含まれる。このため、図 2.6 に示すように、誤差を含む速度、角速度を累積して求めた自己位置には

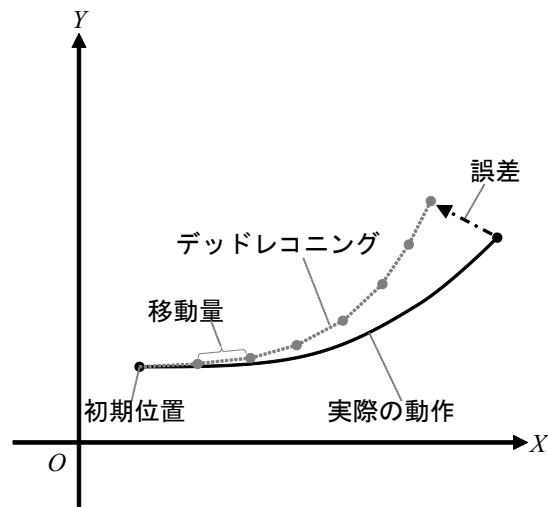


図 2.6: デッドレコニングの誤差の累積

誤差が累積し、次第に誤差が増大する問題がある。また、姿勢に大きな誤差が生じると走行するにつれて位置の誤差が増大するため、姿勢の誤差は致命的な問題となる。そこで、誤差を修正するために 2.1.3 節にて後述する地図上の絶対位置推定と並用して位置を推定する。

#### 生活環境におけるデッドレコニングの誤差

デッドレコニングは、実空間上での自己位置を推定する方法であり、自律ナビゲーションの自己位置推定においては、地図の歪みは十分小さいとして適用している。一般的にデッドレコニングの誤差とは、実空間での自己位置の推定誤差を意味するが、自律ナビゲーションの自己位置推定では、地図上の自己位置の推定誤差が問題となる。図 2.5 に示すように、地図上の自己位置の推定誤差は、実空間上の実際の自己位置に依存しない。すなわち、実空間上の推定誤差に関係なく、地図の歪みとデッドレコニングによる推定によって決まる。しかし、実空間上の自己位置の推定誤差を軽減すると、結果として地図上の自己位置の推定誤差を小さくなると期待されるため、ここでは、実空間上の自己位置の推定誤差と地図上の自己位置の推定誤差の両方について述べる。

初めに、実空間での自己位置の推定誤差について述べる。デッドレコニングは地図作成における自己位置推定にも用いられる。このため、実空間上の自己

位置の推定誤差を軽減すると、同時に地図の歪みを小さくすることが可能と期待できる。このことから、結果として地図上の自己位置の推定誤差を小さくなると期待される。

2次元平面上のデッドレコニングでは、実空間を水平な平面に近似するため誤差が生じる。実際の路面は水平ではなく、傾斜や凹凸があるため、路面の素材や形状がこの誤差の原因となる。生活環境においては、屋外の舗装路面などは水はけのために傾斜が設けられていたり、オフィス環境では路面にカーペットが敷かれているなど、この誤差が生じる環境は広く存在する。この誤差に対して、[17]では、ジャイロセンサを用いて姿勢の推定精度を向上することで、実空間におけるデッドレコニングの推定精度を向上し、同時に歪みの小さい地図を作成している。しかし、生活環境中では、カーペットや坂などを走行する際には位置の推定に誤差が生じることもあり、様々な環境に対して、実空間上でのデッドレコニングの誤差を軽減することは困難である。

地図上の自己位置の推定誤差について述べる。自律ナビゲーションにおけるデッドレコニングでは、実空間、厳密には実空間を近似した水平な平面上でのロボットの動作と地図の空間上での動作が十分近いものとして、地図上の自己位置を推定している。しかし、実際には、2.1.1節で述べたように、実空間と同様とみなせるほど歪みの小さい地図が作成できるとは限らない。局所的に歪みが小さい地図を用いる場合、長い距離を走行すると誤差が大きくなることがある。地図上の自己位置の推定誤差は、地図の歪みとデッドレコニングによる推定によって決まる。ここで、地図の歪みは地図作成時の自己位置推定の誤差に依存する。デッドレコニングの推定誤差に再現性がある場合、地図の作成時の自己位置推定もデッドレコニングとすることで、地図上の自己位置の推定誤差を軽減できることがある[16]。つまり、地図の作成時の自己位置推定もデッドレコニングとすることで、地図の空間上でのロボットの動作はデッドレコニングによる推定と同様に歪み、誤差が小さくなる。ただし、地図の歪みは環境、すなわち地図の作成時におけるセンサの計測に依存するため、地図の作成時と同じ経路を走行する場合にのみ適用可能な方法となる。また、デッドレコニングによる地図の作成では、ループする経路や往復する経路のランドマークの位置

が曖昧となりやすいことも問題となる。

### 2.1.3 地図上の絶対位置推定

地図上の絶対位置推定(以降, 簡単のため絶対位置推定と記述する)では, 外界センサの計測をもとに自己位置を推定する。ロボットの自己位置が正確であれば, 外界センサの計測から得られるランドマークと地図の環境の情報から得られるランドマークが一致することが期待できる。絶対位置推定について図 2.7 に示す。ロボットには (a) に示すようにランドマークが記された地図を与える。(b) に示すようにセンサにより 2 番のランドマークが観測された時, ロボットの位置は地図上の 2 番のランドマークの前であることが分かる。絶対位置推定は, 計測の特徴と地図が一致するロボットの位置, 姿勢を推定する課題である。デッドレコニングがロボットの動作をもとに, 既知の初期位置からの相対的な位置を推定することに対して, 絶対位置推定では, 地図上の絶対位置を推定する。このため, 絶対位置推定では誤差が累積しない。しかし, 多く方法は計算負荷が重く, 推定に 1 秒以上かかることもある。このため, 一般的には, 次に絶対位置推定により自己位置を推定するまでの間は, デッドレコニングにより自己位置を推定する。同時に, 絶対位置推定はデッドレコニングの誤差を修正するために用いられる。

自己位置の追跡における絶対位置推定では, 多くの場合, 直前の自己位置などを初期位置として, センサの計測と地図が一致するロボットの位置, 姿勢を探索する。この時, 多くの絶対位置推定は, 直前の自己位置の誤差が小さいと仮定し, 現在の自己位置は直前の自己位置の付近であるとして推定を容易にし, また誤りを少なくしている。このため, 絶対位置推定ではデッドレコニングの誤差を小さくすることが, 1 つの課題となる。

#### 地図のランドマーク

ランドマークとして用いられる特徴や地図の表現方法は, 使用するセンサや手法により様々であるが, 一般的に, 建物の壁の様な静止物から得られる特徴

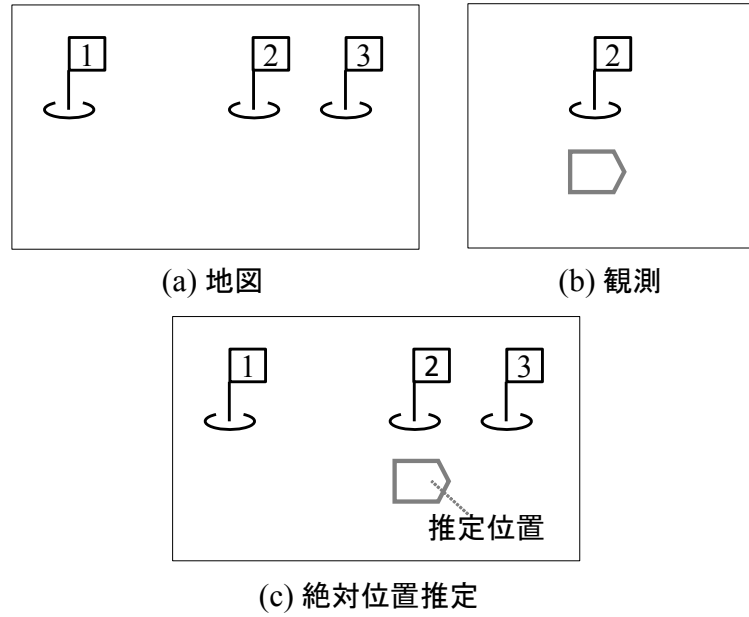


図 2.7: 地図上の絶対位置推定

が、有意となる。ロボットの位置、姿勢が十分近い場合、静止物などは同様に計測されることが期待できるため、地図と計測で特徴が一致することが期待できる。地図の作成方法には、環境を走行した際のセンサの計測から作成する方法と、人が記述する方法がある。前者は、事前にロボットを操作するなどして、環境中を走行しながらセンサの計測を収集する。この時のセンサの計測よりランドマークとなる特徴とその位置を取得し、地図に環境の情報として記述する。後者は、人がランドマークとなる特徴とその位置を記述する方法である。この方法では、人の負担が大きくなるため、特定の物体の位置などの簡単に表現できる地図以外の適用は難しい欠点がある。

ここで、地図の環境の情報は、一般的に特徴と位置によって表現され、その表現方法は特徴ベースと位置ベースに分けられる [18]。特徴ベースでは、ある特徴に対してその位置を記述する。代表的な例としては、点群で表現された地図があり、以下の様に地図  $M$  を点の集合として記述できる。

$$M = \{m_1, m_2, \dots, m_N\} \quad (2.1)$$

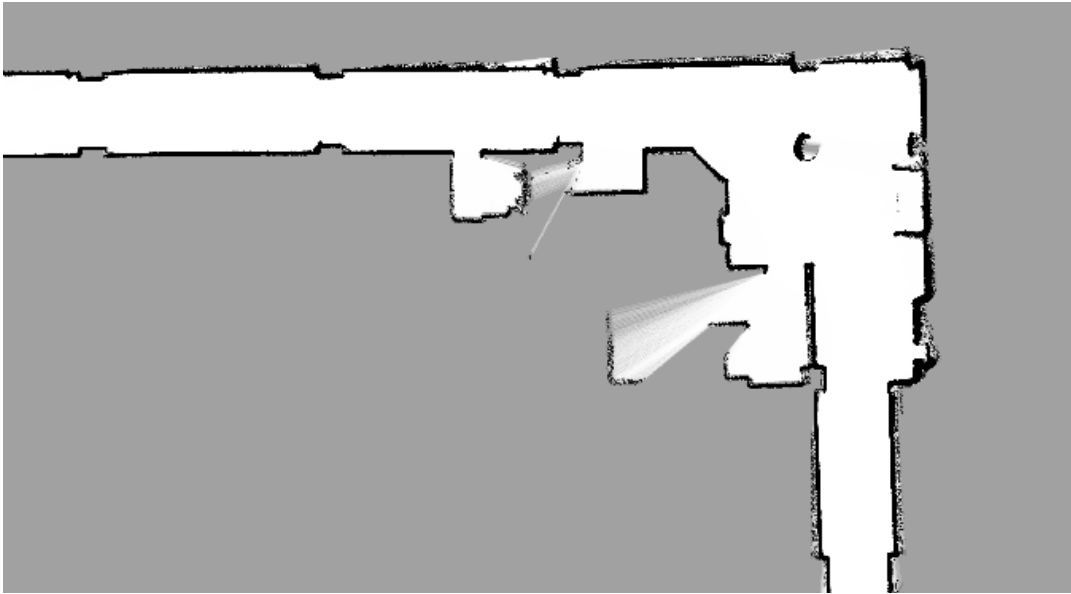


図 2.8: 占有格子地図の例

$$m_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix} \quad (2.2)$$

位置ベースでは，ある位置に対して，その位置の特徴を記述する．位置ベースの地図は，均等な格子状に区切られた地図に対して各セルにその位置の特徴を記述したり，位置に対する関数で表現される．また，特徴がスカラー量で表現できる場合は，地図を画像として表現できる．位置ベースの地図では位置をインデックスとして，特徴を参照することができる．代表的な例としては，占有格子地図があり，図 2.8 に示すように，各ピクセルの位置に対して，静止物により占有されている確率を記述する．この例では，占有されている確率が高いほど黒く示している．

#### 地図と計測の対応付け

絶対位置推定では，自律走行時のセンサの計測と地図が一致するロボットの位置，姿勢を求める．センサの計測と地図が一致するロボットの位置，姿勢を求めるには，センサの計測と地図の一致度の評価関数を用意し，これが最大となる位置，姿勢を正しい位置，姿勢とし，最適化手法を適用して求める方法が

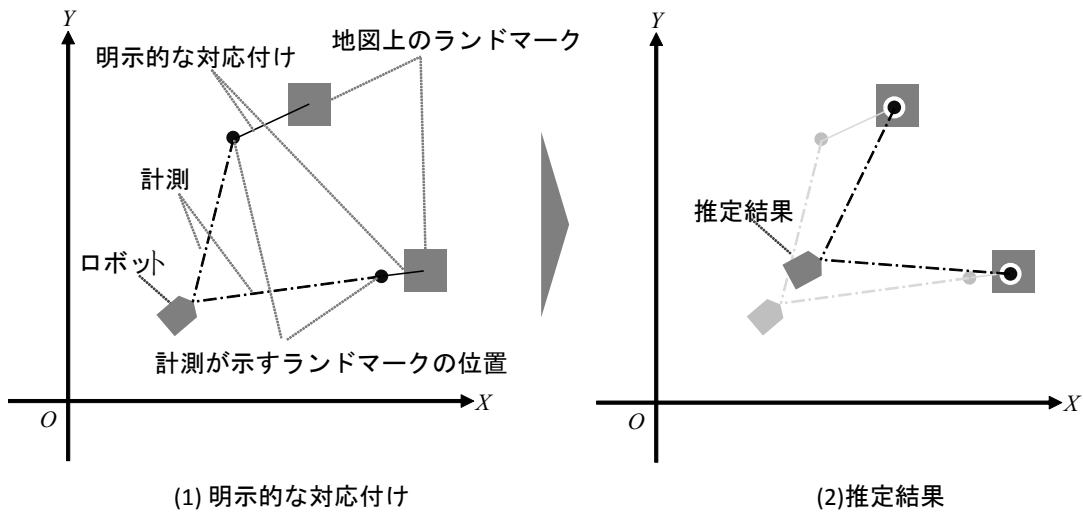


図 2.9: 明示的な対応付け

ある．また，ランドマークに点や線などの明示的な形状で表現できる特徴を用いる場合は，幾何学的にロボットの位置，姿勢を求めることもでき，三角測量がこれにあたる．ここで，計測の特徴に対して，地図上で同一のものを示す特徴を見つけることが必要となる．本論文では，この計測の特徴と地図の特徴で同一のものを示す組み合わせを見つける操作を対応付けと呼ぶ．

対応付けの方法の1つは，図 2.9 に示すように地図と計測の特徴を明示的に対応付ける方法である．明示的な対応付けは，十分に近い特徴同士を明示的に結びつける方法である．この方法は位置ベースの地図でも適用可能だが，多くの場合，特徴ベースの地図を用いて地図に記述された特徴の中から，計測された特徴に近い特徴を検索することで実現される．図 2.9 に例を示す．図 2.9 では計測の特徴を，その位置から最も距離が近い地図上の特徴と対応付け，対応付けられた特徴が一致するロボットの位置，姿勢を推定結果としている．このような明示的な対応付けの例として，ICP スキャンマッチング [19][20] があり，ICP スキャンマッチングでは自律走行時の計測された特徴(点)を地図上の最も距離が近い特徴に対応付ける．

これに対して，計測の特徴と同じ位置の地図の特徴を暗黙のうちに対応付ける方法がある．位置ベースの地図を用いる場合，計測の特徴の位置をインデックスとして地図を参照すればよいため，対応付けが容易である．ここで，計測



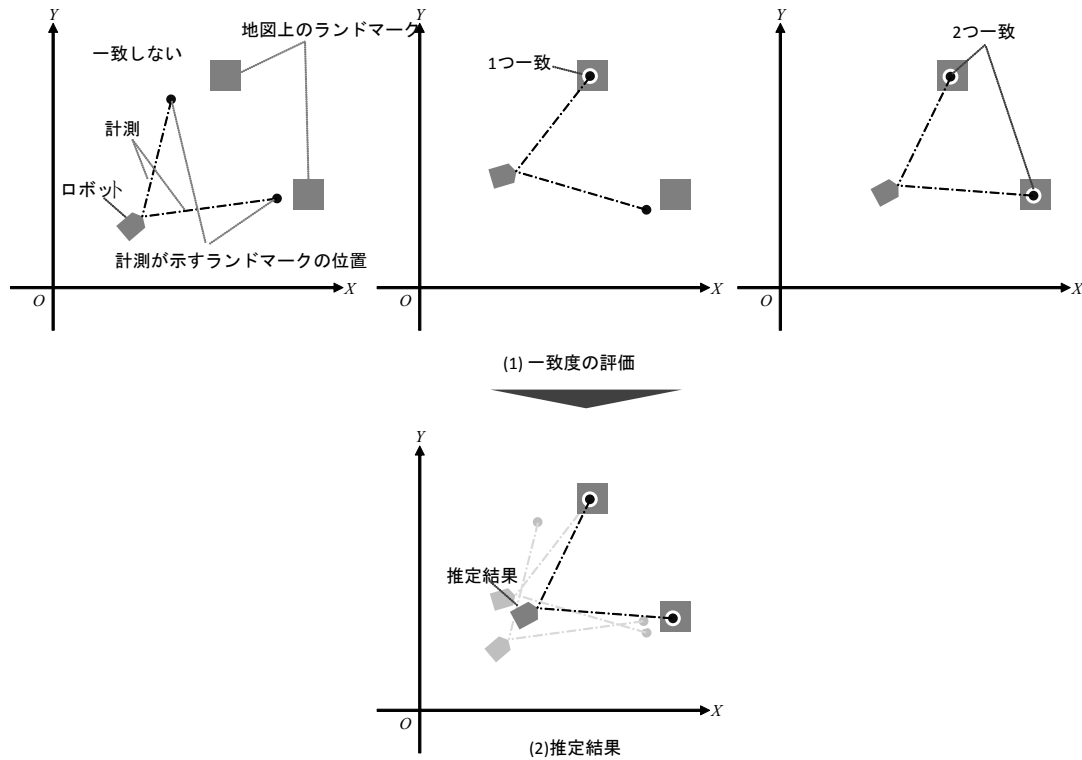


図 2.10: 暗黙の対応付け

の特徴の位置は、直前の時刻の自己位置の推定値などのロボットの位置、姿勢の候補をもとに計算される。この位置、姿勢の候補には誤差があると考えられるため、計測の特徴と地図の特徴は重ならないことがある。このため、何らかの方法でロボットの位置、姿勢を探索する。代表的な例としては、モンテカルロ法 [21] による最適化がある。図 2.10 に例を示す。図 2.10 では、地図には各位置に対するランドマークの有無が記載されており、計測の特徴の位置における地図の特徴の有無により、一致度を評価している。この一致度の評価値が最大となる位置、姿勢を自己位置としている。モンテカルロ法を用いた自己位置推定 (MCL: monte calro localization) はパーティクルフィルタによる自己位置推定 [22] として知られる。パーティクルフィルタについては詳細を 2.1.4 節にて後述する。

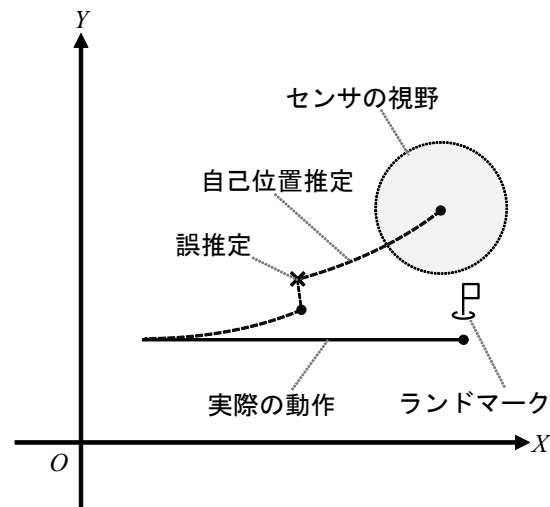


図 2.11: 誤推定による自己位置推定の破綻

### 誤推定

自己位置の追跡では、絶対位置推定の誤り（以降、本論文では誤推定と呼ぶ）が致命的な問題となる。多くの絶対位置推定の手法は、直前の自己位置の誤差が小さいと仮定している。ここで、絶対位置推定が誤推定を起こした場合、誤差が増大することが考えられる。また、デッドレコニングでは、誤差を修正することはできない。このため、図 2.11 に示すように、もう一度絶対位置推定により自己位置を推定しても、直前の自己位置の誤差が大きく、再び誤推定を起こす、あるいは推定できないことがある。図 2.11 に示す例では、自己位置の誤差が増大したため、ランドマークとなる特徴を見失い絶対位置推定に失敗している。このように誤推定により誤差が増大すると、自己位置の追跡の枠組みでは自己位置を修正することは困難であり、誤推定により自己位置が推定できない状況（自己位置推定の破綻）に陥ることがある。

生活環境における誤推定の原因として、不安定な計測があげられる。生活環境においては、ロボットの位置、姿勢が十分近い場合においても、しばしばセンサの計測が異なる特徴を示すことが起こる。これは、歩行者などの時間による変化や、日照やロボットの振動など計測の条件により、センサの計測がバラつき、不安定となることが原因である。このような不安定な計測は偽のランドマークとなり、誤推定を引き起こす。ここで、本論文では、誤って対応付けられ

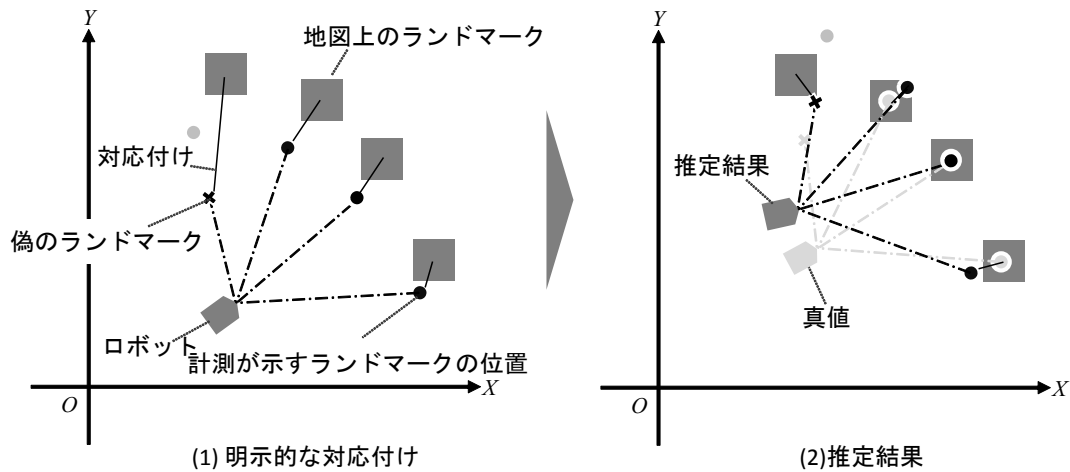


図 2.12: 偽のランドマークに対する明示的な対応付け

た特徴を偽のランドマークと呼ぶ。この偽のランドマークは、地図作成のために収集した計測、自律走行時の計測の双方における不安定な計測が原因となる。

事前に走行した際のセンサの計測から地図を作成する場合、この時の不安定な計測の特徴が地図に記述されてしまうことがある。不安定な計測はバラつくことがあり、地図に記述されると誤って一致しないはずの特徴と対応付けられることがある。不安定な計測から自己位置を推定することは困難であるため、不安定な計測の特徴は地図に記載されない、または不安定であることを記述していることが好ましい。地図に記載されないと、計測の特徴と誤って対応付けることがなく、誤推定を防ぐことができる。このため、地図の作成では地図を作成する際にセンサの計測より、どのようにランドマークとなる特徴を取得するかが課題となる。あるいは、計測値がバラつくことを地図を表現できるように、環境の情報の表現方法を工夫することも考えられる。

偽のランドマークについて、対応付けの際に自律走行時の計測上の歩行者などの地図上に一致する特徴のないものをどのように処理するかが問題となる。特徴を明示的に対応付ける方法では、誤って本来一致しない特徴を対応付けた場合、誤推定を起こすことがある。図 2.12 に示すの例では、偽のランドマークを誤って対応付けたため、この組み合わせが近くなる位置、姿勢を推定結果とし、誤推定を起こしている。このような明示的な対応付けの誤りについては、直前の自己位置の推定誤差に反して距離が大きく離れている組み合わせは、誤った

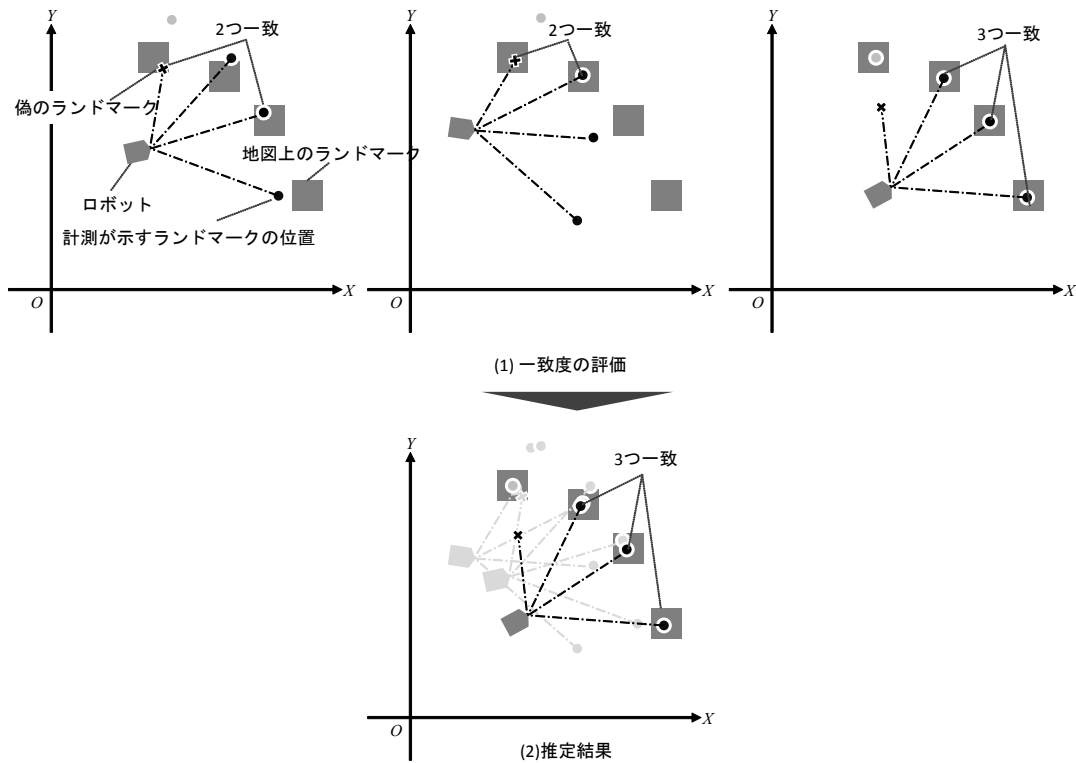


図 2.13: 偽のランドマークに対する暗黙の対応付け

対応付けと判断することで対処できる．この方法は，直前の自己位置の推定誤差が大きいと機能しないため，デッドレコニングの誤差を小さくすることが求められる．

暗黙に対応付ける方法は，偽のランドマークに対して有効であると考えられる．図 2.13 に示すように暗黙に対応付ける場合は，真値付近のロボットの位置，姿勢では偽のランドマークは一致しないが，他のランドマークが一致するため，一致度の評価は高いことが期待できる．このため，正しい位置，姿勢を推定可能と期待できる．しかし，明示的，暗黙，どちらの対応付けにおいても，大きな特徴が偽のランドマークとなった場合，例えば図 2.12，2.13 において 3 つのランドマークが偽のランドマークとなる場合，自己位置推定は困難となる．生活環境においては，催事における人混みやロボットの傾きや路面の傾斜により，計測の大部分が不安定となることがある．このような場合，直前の自己位置の推定誤差をもとに，対応付けや推定結果の誤りを判別することが有効であり，デッドレコニングの誤差を軽減することが，誤推定の問題について有効だと考えら

れる．

### ランドマークの不足

絶対位置推定には信頼できるランドマークが必要となるが、生活環境では、偽のランドマークの問題に加えて、ランドマークが不足する問題がある．これは、歩行者などに遮られ、ランドマークの観測を妨げられる、そもそも、環境中にランドマークとなる特徴が十分でないことが原因となる．ランドマークが不足すると絶対位置推定により自己位置が推定できないため、デッドレコニングの誤差が累積する．このような状況が長く続くと、誤差が増大して自己位置推定が破綻することがある．

この問題に対して、環境中の任意の位置に人工的なランドマークを設置することで、自己位置推定を実現した事例がある [5],[9]．[5]では、ランドマークとしてレーザーフレクタを設置し、センサにより観測し、三角測量により自己位置を推定している．[9]では、無線IDタグを床面に埋め込み、これをランドマークとして位置を得る．このように環境を整備することで、安定して計測可能で、センサの計測において他の特徴と明確に見分けることができるランドマークを用意し、自己位置推定を容易とすることができる．しかし、生活環境においては環境を整備するには工事を要し、コストが高くなったり、人に合わせた環境とするため適用できないことがある．このため、本研究では、環境を整備せずに適用可能な自己位置推定の手法について検討する．

この問題に対しては、2つの対応が必要と考えられる．1つは、様々な環境で容易に観測できるランドマークによる絶対位置推定である．地図上の自己位置推定では、誤差を修正するため地図上の絶対位置を推定することは不可欠である．このため、容易に観測できるランドマークによる絶対位置推定を用いて、様々な環境で誤差を修正できることが求められる．もう1つは、デッドレコニングの精度である．環境中に十分なランドマークがない問題は、環境に依存するため、完全な解決は困難である．そこで、環境中に十分なランドマークがない区間を通過する間に、自己位置推定が破綻するほどの誤差が累積しないよう

に、デッドレコニングの精度が求められる。

#### 2.1.4 自己位置の融合

上述の絶対位置推定において、1回のセンサの計測では自己位置を定められない、または十分な精度で推定できない場合が考えられる。例えば、一様な長い廊下のような環境では、人間でもその場の様子から位置を特定できないことが考えられる。任意に人工的なランドマークを設置する場合、一意に位置を推定できるようにランドマークを配置する事も可能であるが、生活環境のようなあるがままの環境においては、1回のセンサの計測では自己位置を定められない場合が考えられる。

この問題に対して、デッドレコニングと絶対位置推定、時間を経て得られた計測による絶対位置推定結果などの、複数の推定結果を融合することで、より正確に位置を推定する手法が研究されている [18][23][24]。これらの多くは、ベイズの理論をもとにしたベイズフィルタにより、自己位置推定の推定誤差をもとに確率的に融合する手法(確率的自己位置推定)である。これらの手法では推定誤差を考慮することで、自己位置を1つの推定値ではなく、確率分布で表現する。確率分布とは、地図の空間でとりえる自己位置の推定値に対する信頼度であり、これを数学的に表現することで、異なる複数の推定結果からより信頼度の高い自己位置を計算できる。

確率的自己位置推定では、動作モデルと計測モデルから確率分布を考える。動作モデルとはロボットの移動による位置、姿勢の遷移モデルであり、デッドレコニングがこれにあたる。計測モデルとは、環境で取得されるセンサの計測値もしくは特徴の生成モデルであり、実際には地図に記された環境の情報から、あるロボットの位置、姿勢でえられるセンサの計測値や特徴を考えるために用いられる。絶対位置推定では、この計測モデルと実際のセンサの計測値もしくは特徴が一致する位置、姿勢を推定している。ここでは、本研究で用いた拡張カルマンフィルタとパーティクルフィルタによる自己位置の追跡と、確率的自己位置推定の問題として誤差のモデルとパラメータについて述べる。

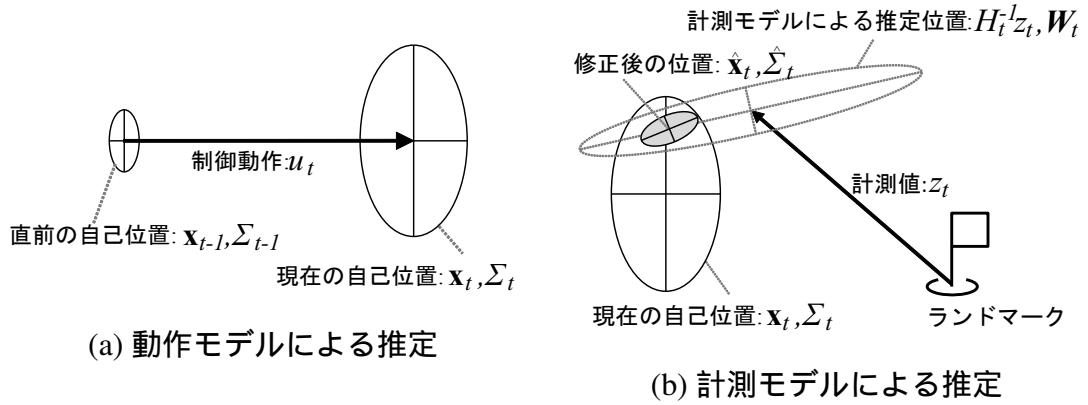


図 2.14: 拡張カルマンフィルタによる自己位置推定

### 拡張カルマンフィルタ

図 2.14 に拡張カルマンフィルタによる自己位置推定について示す．拡張カルマンフィルタによる自己位置推定では，自己位置の推定誤差は正規分布に従うと仮定し，動作モデル，計測モデル，融合結果の確率分布を推定誤差をもとにした正規分布で表現する．そこで，拡張カルマンフィルタにおいては，それぞれの推定結果の自己位置とその誤差の共分散行列を求める．

#### 1. 動作モデルによる推定

動作モデルによるロボットの自己位置  $\mathbf{x}_t = (x_t, y_t, \theta_t)^T$  の推定と，この時の誤差分散  $\Sigma_t$  を以下に示す．

$$\mathbf{x}_t = g(\mathbf{x}_{t-1}, \mathbf{u}_t) \quad (2.3)$$

$$\Sigma_t = \mathbf{G}_t \Sigma_{t-1} \mathbf{G}_t^T + \mathbf{V}_t \Sigma_u \mathbf{V}_t^T \quad (2.4)$$

動作モデル  $\mathbf{x}_t = g(\mathbf{x}_{t-1}, \mathbf{u}_t)$  は制御動作  $\mathbf{u}_t$  による自己位置の遷移を示す．動作モデルにはしばしばデッドレコニングが用いられ，この時， $\mathbf{u}_t$  は内界センサより得られた速度，角速度や移動量となる． $\mathbf{G}_t$  は  $\mathbf{x}_t$  に関する  $g$  のヤコビ行列， $\mathbf{V}_t$  は  $\mathbf{u}_t$  に関する  $g$  のヤコビ行列， $\Sigma_u$  は  $\mathbf{u}_t$  の誤差の共分散行列である．

#### 2. 計測モデルによる更新

計測値  $z = h(\mathbf{x}, M)$  は，地図  $M$  と計測モデル  $h$  とから考えられるロボット

の位置，姿勢が  $\mathbf{x}$  である時のセンサの計測値を示す．ただし，自己位置を推定する場合， $z$  は現時のセンサの計測より得られる定数であり，位置，姿勢  $\mathbf{x}$  が変数となり，計測モデル  $z = h(\mathbf{x}, M)$  を満たす， $\mathbf{x}$  を求める事になる．計測モデルによる更新を以下に示す．

$$\hat{\mathbf{x}}_t = \mathbf{x}_t + \hat{\Sigma}_t \mathbf{W}_t^{-1} (\mathbf{H}_t^{-1} z_t - \mathbf{x}_t) \quad (2.5)$$

$$\hat{\Sigma}_t = (\Sigma_t^{-1} + \mathbf{W}_t^{-1})^{-1} \quad (2.6)$$

$\mathbf{H}_t^{-1} z_t$  はセンサの計測より推定される自己位置， $\mathbf{W}_t$  はその誤差の共分散行列である．ここで， $\mathbf{W}_t^{-1} = \mathbf{H}_t \mathbf{Q}_t^{-1} \mathbf{H}_t^T$  であり， $\mathbf{H}_t$  は  $\mathbf{x}_t$  に関する計測モデル  $h$  のヤコビ行列， $\mathbf{Q}_t$  は計測値の誤差の共分散行列である．拡張カルマンフィルタによる自己位置推定では，この式 2.5，2.6 によってランドマークが観測されるたびに自己位置の更新を行う．

## パーティクルフィルタ

パーティクルフィルタによる自己位置の推定では，ロボットの位置，姿勢の候補を多数生成する．このロボットの位置の候補をパーティクルと呼び，時刻  $t$  における  $i$  番目のパーティクルを  $\mathbf{x}_{t,i} = (x_{t,i}, y_{t,i}, \theta_{t,i})$ ，パーティクルの集合を  $\chi_t = \{\mathbf{x}_{t,i}\}$  と示す．パーティクルフィルタでは，このパーティクルの集合の分布により確率分布を表現する．

### 1. 動作モデルによる推定

パーティクルフィルタでは，動作モデルに従って各パーティクルを遷移させる．この時， $\mathbf{u}_t$  にランダムな値  $\varepsilon_{t,i}$  を加える．これは，動作モデルにおける誤差を反映するためであり， $\varepsilon_{t,i}$  は  $\mathbf{u}_t$  の誤差の分布に応じて生成する．

$$\mathbf{x}_{t,i} = g(\mathbf{x}_{t-1,i}, \mathbf{u}_t + \varepsilon_{t,i}) \quad (2.7)$$

### 2. 計測モデルによる重み付け

各パーティクルに対して計測値  $z_t$  をもとに重みづけをする．各パーティク



ルの重み  $w_{t,i}$  は以下となる．

$$w_{t,i} = p(z_t | \mathbf{x}_{t,i}, M) \quad (2.8)$$

$p(z_t | \mathbf{x}_{t,i}, M)$  はロボットの位置が  $\mathbf{x}_{t,i}$  のときに，計測値  $z_t$  が得られる尤度を示す．これは，ロボットの位置，姿勢が  $\mathbf{x}_{t,i}$  である場合に，センサの計測値として  $z_t$  が，地図  $M$  に対してどれくらい尤もらしいかを示す値であり，地図と計測値の一致度をもとに計算する．この重みは計測モデルによる推定の確率分布を表現している．

### 3. リサンプリング

リサンプリングではパーティクルの重み  $w_{t,i}$  をもとに，パーティクルの集合  $\chi_t$  から新たなパーティクルの集合を作成する．パーティクルの集合  $\chi_t$  から任意の数のパーティクルを確率的に選択し，この複製を新たなパーティクルの集合とする．この時，パーティクルが選択される確率は，パーティクルの重み  $w_{t,i}$  に比例し，重複して選択されてもよいとする．この操作により，動作モデルにより遷移したパーティクルの分布に，計測モデルによる推定が反映される．パーティクルは動作モデルとその誤差に従って遷移し，計測モデルによる重みが高いパーティクルが多く残り，一方，重みが弱いパーティクルは淘汰されることとなる．

最終的なロボットの位置，姿勢には，パーティクルの位置，姿勢の重み付き平均や，最も重みの強いパーティクルの位置，姿勢が用いられる．

### 誤差のモデルとパラメータ

確率的自己位置推定では，動作モデルと計測モデルの誤差を考慮することで，より信頼度の高い自己位置を推定している．この誤差のモデルは拡張カルマンフィルタでは正規分布で近似されており，パラメータとして制御動作の誤差  $\Sigma_u$ ，計測値の誤差  $Q_t$  が設けられている．パーティクルフィルタにおいても，一般的に制御動作の誤差  $\varepsilon_{t,i}$  は正規分布に従うとして実装される [24]．また，誤差の分布は既知であるとして，これらのパラメータは経験的に決められた固定値が与

えられる。しかし、実際には動作モデル、計測モデルともに環境に依存して誤差が生じることが考えられる。環境に依存する誤差は、系統誤差であり、同様に生じる誤差であることが考えられる。このような系統誤差の分布は正規分布と異なるため、系統誤差が大きくなる環境では、正規分布に近似した確率分布が不適切となり、自己位置推定の精度が低下することが考えられる。

## 2.2 生活環境における自己位置推定の課題

自己位置の追跡の方式においては、自己位置推定を破綻させないように誤差の増大を防ぐことが求められる。生活環境において誤差を増大させる原因として、2.1 節で述べた自己位置推定の方法とその生活環境における問題より、以下の2点が課題としてあげられる。

- 絶対位置推定における誤推定への対策
- ランドマークの不足により絶対位置推定が不可能な状況への対策

結論として、これらの課題の解決には、絶対位置推定とデッドレコニングの改善が必要と考えられる。ここでは、それぞれの課題について整理し、生活環境における自己位置推定における絶対位置推定とデッドレコニングの課題をまとめ、次に、絶対位置推定とデッドレコニングのそれぞれについて、具体的なアプローチを検討する。

### 2.2.1 地図上の絶対位置推定における誤推定への対策

誤推定の原因は、歩行者など時間による変化や、日照やロボットの振動など計測条件の変化による不安定な計測である。不安定な計測が地図上か自律走行時の計測のどちらか、あるいは両方において、偽のランドマークとなる事で誤った位置、姿勢を推定してしまう。不安定な計測と偽のランドマークの問題に対する検討として以下の項目が考えられる。

- 計測方法

誤推定の問題を解決する単純な方法は、根本的な原因である不安定な計測が生じないようにセンサおよび計測方法を工夫することである。生活環境においては、不安定な計測の主な原因は人の活動と考えられ、人の活動の影響が少ない高所を計測することで改善が期待できる。例として、屋内環境限定であるが、天井の画像からランドマークを取得して自己位置を推定する方法が研究されている [9][25][26]。

- 地図

地図に記述された不安定な計測の特徴が、誤って対応付けられないようにする。これには2つの方法が考えられる。1つは地図から不安定な計測の特徴を除外する、これにより、地図上の偽のランドマークと計測の特徴を誤って対応付けすることを防ぐことができる。この方法については、地図作成の工夫、または、人手による地図の編集が考えらる。もう1つは地図の記述を工夫し、環境の情報として計測のバラつきを表現する。これにより、計測がバラつくことを考慮して、地図と計測の一致度を評価できる。

- 対応付け

自律走行時の計測の特徴を対応付けの際、もしくは対応付け以前に不安定な計測の特徴を除外した上で、自己位置を探索する。これには3つの方法が考えられる。1つ目はランドマークとなる特徴を抽出する過程で除外する方法である。画像中の白線などの抽出が容易な特徴をランドマークとして用いる場合、あらかじめ不安定な計測の特徴を除外することが期待できる。

2つ目は、自己位置の推定精度の向上である。この方法では対応付けの際に、直前の自己位置の確率分布に反して遠くの特徴に対応付けられた場合、誤りだと判定することができる。誤推定は絶対位置推定の問題であり、この方法はデッドレコニングの精度向上により実現する。また、同時にデッドレコニングの確率分布を適切に表現する必要がある。

もう1つは、パーティクルフィルタによる自己位置推定を適用するなどし

て，暗黙の対応付けにより自己位置を推定する方法である．

これらの不安定な計測の対策の他に，誤推定を検出することが考えられる．これは，対応付けにおける対策と同様に直前の自己位置の精度に反して，遠く的位置が推定された場合，誤推定と判定する．これにより誤推定の結果を無視することができる．この方法においても，デッドレコニングの精度向上し，確率分布を適切に表現することが有効と考えられる．

### 2.2.2 ランドマークの不足への対策

生活環境においては，歩行者にセンサの視界を遮られるなどしてランドマークが観測できないことや，そもそもランドマークとなる特徴が不足する区間がある．ランドマークが不足している状況では，デッドレコニングのみで自己位置を推定することとなり，誤差が累積し，増大してしまう．ランドマークの不足により絶対位置推定が不可能となる状況については以下の2点について検討する必要がある．

- ランドマークとする特徴

この問題に対する単純な対策は，様々な環境で容易に計測可能である特徴をランドマークとして絶対位置推定することである．

- デッドレコニングの精度向上

様々な環境で容易に計測可能である特徴を用いることで改善できるが，ランドマークとなる特徴が存在するかは，環境に依存するため，根本的な解決はできない．ランドマークが不足し，絶対位置推定が不可能な状況において，誤差の増大を防ぐにはデッドレコニングの精度を向上する必要がある．デッドレコニングの精度を向上することにより，長い区間でランドマークが不足する環境においても，自己位置推定の破綻を防ぐことが期待できる．

### 2.2.3 地図上の絶対位置推定の検討

絶対位置推定に求められる課題は、不安定な計測が生じない計測方法およびセンサ、もしくは地図の作成および対応付けにおける不安定な計測への対策である。また、ランドマーク不足の問題に対しては、走行環境にいて容易にランドマークが観測できることが求められる。絶対位置推定は様々なセンサを用いた手法が多数研究されている。そこで、既存の手法に対して、これらの課題の解決について検討した。

ここでは、既存の絶対位置推定として、移動ロボットの自己位置推定に広く用いられるレーザスキャナなどから得られる形状特徴を用いた手法と、カメラによる画像を用いた手法について検討する。レーザスキャナやカメラなど、形状特徴、画像を計測するセンサは、高速で大量の情報を取得できるため、ロボットの自己位置推定に一般的に用いられる。また、GPS についても検討する。GPS は地図とセンサデータの対応付けを必要としないため、地図上の絶対位置推定とは異なる。しかし、地球上における絶対位置を計測できることから、同様の役割で用いられるため、検討することとした。以降、形状特徴を用いた手法、画像を用いた手法、GPS についての検討について述べる。

#### 形状特徴を用いた絶対位置推定の検討

環境の形状を取得するセンサとして、レーザスキャナ、ミリ波レーダ、ステレオビジョン等があるが、ここでは外界センサにレーザスキャナを用いた場合について検討する。レーザスキャナは自己位置推定の研究に広く用いられている。レーザスキャナはレーザを照射し、その伝搬時間をもとに物体までの距離を計測する。本論文ではレーザスキャナによる計測をスキャンデータとよぶ。視野については、多くのレーザスキャナは180度以上の視野をもち、数メートルから数十メートルまで、広い範囲を計測できる。このため、ランドマークを観測できる機会が多くなる。また、指向性が高く、図2.15(図の左は計測した環境の写真、図の右はその環境におけるスキャンデータ)に例を示すように計測を点群で表現することができる。この点群は、物体までの距離の計測値をもとに得ら

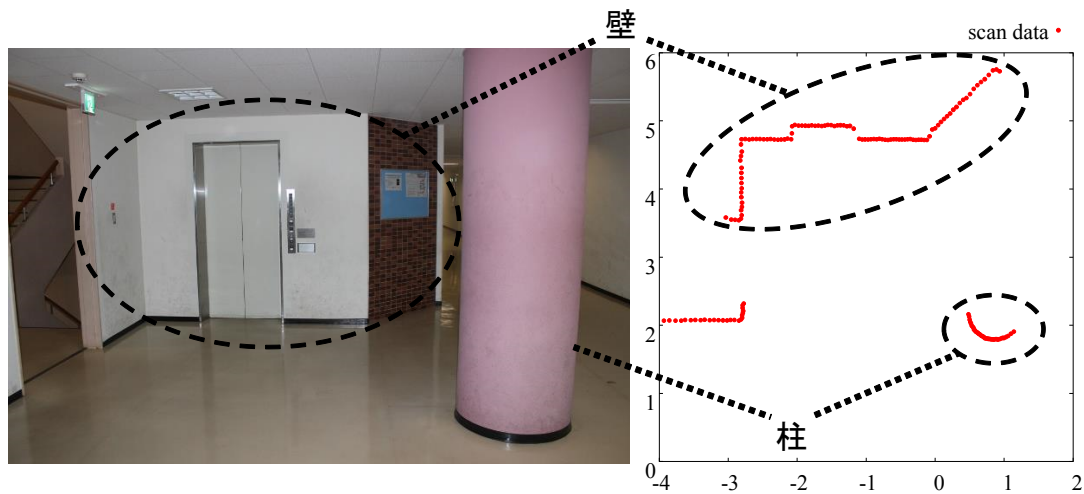


図 2.15: スキャンデータ

れるレーザの反射点であり，物体の表面の位置を示す．視野角度に対する分解能も高いため，周囲の物体の形状を明確に計測できる．能動的にレーザを照射して計測するため周囲の照明条件に左右されず，屋内外で安定した計測が得られることが期待できる．また，高所の形状特徴は，人の活動による影響を受けにくいと考えられるため，計測の方法を工夫することで誤推定の問題を解決できると期待できる．

レーザスキャナによる形状特徴を用いた絶対位置推定では，明示的な形状特徴を用いる手法とスキャンデータ全体の形状特徴を用いる手法がある．

- 明示的な形状特徴を用いる手法

明示的な形状特徴を用いる手法では，人工物の壁などから得られる線分，電柱や街路樹などから得られる円弧，人工物の壁から得られる線などの特定の形状をランドマークとして用いる [27]．スキャンデータから，これらの特徴を抽出し，地図に登録されたランドマークと対応付け，一致する位置，姿勢を求める．スキャンデータから線分などの特徴を抽出したのちに対応付けるため，対応付けにおける不安定な計測の問題に対して有効である．また，地図の作成時にも同様に特徴を抽出することで，地図の作成時の不安定な計測についても解決できる．人工物には直線や円弧の特徴多いため，屋内や市街地において適用可能と考えられる．しかし，公園などの

人工物が乏しい環境では、安定してランドマークが確保できないことがある。

- スキャンデータ全体が示す形状特徴を用いる手法

スキャンデータ全体を用いる手法は、スキャンデータの点群1点1点、あるいは照射したレーザ1本1本をランドマークとして、地図と点群の一致度を評価する方法である。この一致度が最大となるロボットの位置、姿勢を探索することにより、自己位置を推定する。スキャンデータ全体と地図との一致度を評価する方法は様々であり、スキャンデータの点群と地図に記された静止障害物の位置が一致するか評価する方法 [13][17][28] や、地図の静止障害物の位置をレーザが透過していないか評価する方法 [15][29][30] などがある。ランドマークとなる特徴に対してとくに条件がなく、ある程度のスキャンデータが得られると自己位置が推定できるため、ランドマークの不足の問題に対して有効である。しかし、基本的にスキャンデータ全体を用いるため、単純には不安定な計測の特徴を除外できない。このため、生活環境に適用する場合、地図の作成と対応付けの両方で、不安定な計測による誤推定が問題となる。

#### 画像による手法の検討

カメラはロボットのセンサとして広く用いられ、画像の輝度や色情報から様々な特徴が得られる利点がある。自己位置推定においても画像を用いた様々な手法が研究されている。画像から得られる特徴点をもとに3次元の地図と照合する手法 [31][32][33][34] や画像中の模様を照合する方法 [35] が研究されている。しかし、カメラは自己位置推定に使用するセンサとして、生活環境とくに屋外環境において多くの問題が報告されている [17]。屋外では時刻や天候によって画像の見え方が変化し、不安定となる。樹木の枝葉や草の影により複雑な模様が生じ、これらより画像特徴が抽出されるが、これらは季節や時刻によって変化するため、不安定となる。また、屋外のみならず屋内においても窓のそばでは、太陽光のため日中は白とびや黒つぶれが生じたり、夜間においては暗いた

め、そもそも特徴が得られないことが考えられる。このような照明条件による不安定な計測やノイズに対して、白線などの高いコントラストのエッジを特徴を用いて自己位置を推定する研究がされている [17]。特定の特徴をランドマークとするため、明示的な形状特徴をランドマークとする手法と同様に、偽のランドマークの問題に有効と考えられる。ランドマークが観測できる環境については、一般市街地や車道などでは、タイルの模様、車道の白線からエッジが検出可能と期待できるが、公園や建物内などでは十分特徴がえられない区間があると考えられ、多様な環境への適用は難しく、ランドマークの不足が問題となる。

### GPS の検討

GPS による測位は環境中にランドマークを必要としないため、屋外において様々な環境で自己位置が推定可能と期待できる。しかし、実際には、GPS の測位には、経路周辺の樹木や建物が衛星を遮蔽したり、電波の反射や回折によるマルチパスが生じ、測位精度が低下する問題がある。このような問題に対して、測位した位置が系統的にずれることから直進した際の進行方向の推定に用いる方法が研究されている [36]。天頂に向けた赤外全周カメラの情報から得られる遮蔽物の情報や地形の 3 次元の地形情報から、衛星の可視判別し、不可視衛星からの信号を棄却することで精度の低下を解決する手法 [37][38] が研究されている。しかし、前者では、姿勢の推定のみしかできないため、位置の誤差を修正できない。後者では、測位精度が低下する問題は解決できるものの、根本的に電波を受信できる衛星が少なくなり、測位の機会が少なくなる。また、屋内には適用できない。このように、ランドマークの不足と同様の問題がある。

### 地図上の絶対位置推定のアプローチ

本研究では、外界センサにレーザスキャナを用いてスキャンデータ全体が示す形状特徴を用いる枠組みを採用することとした。スキャンデータ全体が示す形状特徴を用いる枠組みでは、ある程度のスキャンデータが得られれば自己位置の推定が可能となるため、ランドマークの不足の課題に対して有効である。一



方，他の方法では，ランドマークが不足し，絶対位置推定が不可能となる環境が広く存在する．画像による手法では，人の活動に加えて，照明条件により計測が不安定となり，屋外環境での適用は難しい．画像における高いコントラストのエッジや明示的な形状特徴を用いる手法では，特定の特徴を必要とするため，ランドマークとなる特徴が乏しい環境では，自己位置が推定できない．また，GPSにおいても，屋内や高い物体により衛星を遮蔽される環境での計測が困難となる．これらの問題は環境に依存する問題であるため，解決は困難である．レーザスキャナを用いてスキャンデータ全体を用いる手法においても，完全な解決は不可能であるが，一般的な生活環境の大部分において適用可能と期待できる．レーザスキャナの計測距離は長い物で数十メートルあり，この範囲内に計測できる物体が存在しない環境は生活環境において稀である．そこで，本研究では，スキャンデータ全体の形状特徴を用いる方法において誤推定の問題を解決する以下の2つの方法を提案する．

#### 1. 高所特徴のスキャンマッチングによる自己位置推定

この方法では，誤推定の問題に対して不安定な計測が生じない様に，計測方法を工夫する．レーザスキャナにおける不安定な計測の主な原因は，人の活動による環境の変化と考えられる．このため人の背よりも高所を計測することで，不安定な計測を防ぐと同時に，人によるランドマークの観測の妨げを防ぐことができる．計測した高所の特徴を用いることで，従来の絶対位置推定でも誤推定の問題の改善が期待される．そこで，レーザスキャナによる計測方法を工夫して，人の背より高所の特徴を計測する．この高所の特徴をもちいてスキャンマッチングの枠組みにより自己位置を推定する．

#### 2. 統計量に基づくレーザスキャナの尤度場を用いた自己位置推定

高所特徴のスキャンマッチングによる自己位置推定では，高所を計測することで人によるランドマークの観測の妨げを防ぐことが期待される，一方，背の高い物体が少ない環境ではランドマークが不足すると考えられる．そこで，計測方法に依存せず誤推定を防ぐため，不安定な計測に口バ

ストな自己位置推定手法を検討した。本手法では、地図の環境の情報として、計測のバラつきを表現するために、反射点観測の尤度を用いる。反射点観測の尤度とは、ある位置で反射点が観測されたときに、その観測がロボットが走行している環境で得られる計測として尤もらしいかを示す値である。本手法ではこの尤度を事前に収集した複数回のスキャンデータの統計量から計算する。複数回のスキャンデータを重ね合わせたとき、静止物の周囲は反射点が密になり、もう1度計測した際に再度静止物の周辺に反射点が観測されることは尤もらしい。一方、不安定な計測が生じた箇所では計測がバラつき、反射点が疎となる。もう1度計測した際に、反射点が疎であった箇所から反射点が観測されることは、静止物の周囲と比較して尤もらしくないと考えられ、尤度が低くなる。この尤度を地図として計測のバラつきを尤度の低さで表現できる。また、自己位置推定においては、提案手法による尤度とスキャンデータの一致度の評価値をパーティクルフィルタにおけるパーティクルの重み付けに適用することで、自己位置を推定する。

#### 2.2.4 デッドレコニングの検討

本研究では、生活環境における自己位置推定の実現のためにデッドレコニングの課題として、精度の向上と誤差のモデルについて検討する。絶対位置推定における誤推定の問題に対しては、デッドレコニングの精度を向上し、誤差のモデルを適切に表現することで、誤った対応付けを防ぎ、また、誤推定の検出することが期待できる。また、ランドマーク不足の問題に対しては、デッドレコニングの精度を向上し、絶対位置推定ができない状況で誤差の累積を軽減することも有効である。そこで、既存の手法に対して、これらの課題の解決について検討した。

ここでは、車輪型移動ロボットのデッドレコニングであるオドメトリの方法とその誤差の考察について述べ、本研究のアプローチについて述べる。

## オドメトリ

本研究で用いる独立二輪駆動型ロボットのオドメトリについて述べる．初めに，ある時刻  $t$  ロボットの位置  $\mathbf{x}_t = (x_t, y_t, \theta_t)^T$  は速度，角速度をもとにロボットの動作から以下で求められる．

$$\begin{aligned}\theta_t &= \int_0^t \omega(\tau) d\tau + \theta_0 \\ x_t &= \int_0^t v_x(\tau) d\tau + x_0 \\ y_t &= \int_0^t v_y(\tau) d\tau + y_0\end{aligned}\tag{2.9}$$

ここで， $\mathbf{x}_0 = (x_0, y_0, \theta_0)^T$  は初期位置を示す．

図 2.16 に独立二輪駆動型ロボットのキネマティクスについて示す．独立二輪駆動型ロボットのオドメトリのキネマティクスパラメータは左右の車輪径  $r_L, r_R$  とトレッド (両車輪間の距離)  $T$  がある．独立二輪駆動型では並進速度  $v$ ，角速度  $\omega$  成分しかもたず，ロボットの向きに対して垂直方向には動かないため， $x, y$  軸それぞれの方向への速度  $v_t^{(x)}, v_t^{(y)}$  は以下となる．

$$\begin{aligned}v_t^{(x)} &= v_t \cos(\theta_t) \\ v_t^{(y)} &= v_t \sin(\theta_t)\end{aligned}\tag{2.10}$$

オドメトリでは，ロボットは水平な路面上を滑らずに走行していると仮定し，並進速度  $v$ ，角速度  $\omega$  を車輪に取り付けたエンコーダにより計測した車輪の回転量とキネマティクスパラメータをもとに推定する．エンコーダにより計測した左右の車輪の回転量をそれぞれ  $\phi_L, \phi_R$ ，エンコーダの計測周期を  $\Delta t$  とすると並進速度  $v$ ，角速度  $\omega$  は以下となる．

$$v = \frac{r_R \phi_R + r_L \phi_L}{2\Delta t}\tag{2.11}$$

$$\omega = \frac{r_R \phi_R - r_L \phi_L}{T \cdot \Delta t}\tag{2.12}$$

オドメトリによる位置の推定では，式 2.9 を離散系で計算するため，漸化式

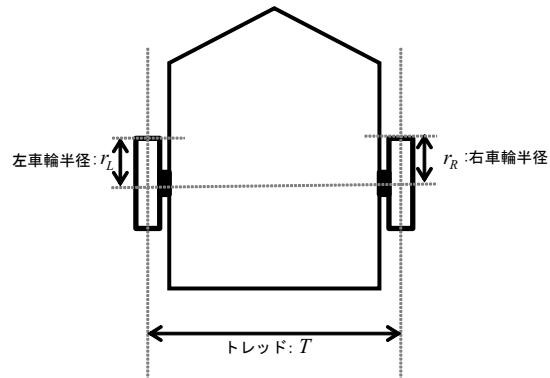


図 2.16: 独立二輪駆動型ロボットのキネマティクスパラメータ

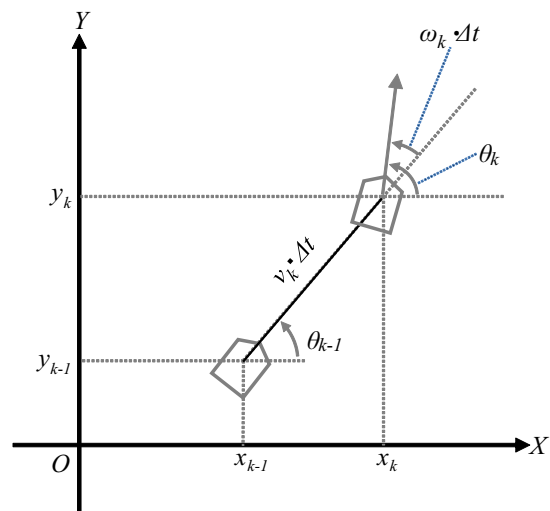


図 2.17: オドメトリによる自己位置推定

に変形し,  $k$  番目の推定値  $\mathbf{x}_k = (x_k, y_k, \theta_k)^T$  は以下の式で求める (図 2.17) .

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \Delta t \cdot \begin{pmatrix} v_{k1} \cos(\theta_{k-1}) \\ v_{k1} \sin(\theta_{k-1}) \\ \omega_{k1} \end{pmatrix} \quad (2.13)$$

### デッドレコニングのアプローチ

実空間上の自己位置推定におけるオドメトリの誤差について述べる．オドメトリの計算より誤差の原因として，キネマティクスパラメータの誤差が考えられる．キネマティクスパラメータは設計図の値や，実測した値が用いられるが，

この値の誤差により，エンコーダより得られる車輪の回転数に対して速度，角速度の推定に誤差が生じる．これはキャリブレーションにより改善することができる [39][40]．また，環境に起因して，具体的には路面に起因して誤差が生じる．オドメトリは滑らかな水平な平面を走行していると仮定しているが，実際には路面には傾斜があったり，素材によっては滑らかではない．このため，路面の傾斜や素材などにより近似による誤差が生じる．この誤差は環境に依存して様々な誤差が生じるため，対応が困難となる．また，スリップなど車輪が滑った際にも誤差が生じる．

地図上の自己位置推定では地図の歪みにより誤差が生じる．これは2.1.2節で述べたように，自律ナビゲーションによる自己位置推定は，ロボットの持つ地図の空間での自己位置の推定に対して，オドメトリは実空間での位置を推定するためである．ここで，実空間上の自己位置の推定誤差を軽減することで，同時に地図の歪みを小さくし，地図上の自己位置の推定誤差が軽減されることが期待できるが，上述のように路面の傾斜などに起因する誤差を軽減することは困難となる．

ここで，オドメトリの誤差は系統誤差と偶然誤差に分けられる．系統誤差は同様に走行すれば，同じ傾向で生じる誤差である．路面の傾斜や素材，地図の歪みに起因する誤差は，同じ場所を走行すると同様の誤差が生じるため，系統誤差と考えられる．偶然誤差は凹凸のはげしい路面や急加速によるスリップ等が要因である．偶然誤差の予測や計測は困難であるが，一方，系統誤差は同様に走行した際に同じ傾向で生じるため，事前に走行した際に生じた誤差をもとに自律走行時に生じる誤差を推定し，修正できると考えられる．また，系統誤差を軽減する事により，従来の実装でも適切に誤差の分布を表現可能と考えられる．図2.18に，同じ軌跡を数回走行した場合のオドメトリの推定とその誤差のイメージ図を示す．系統誤差が大きい場合，誤差の分布を正規分布で近似すると，本来の誤差の分布に対して不適切に大きな分布となる．これに対して，系統誤差を軽減することで，正規分布により誤差の分布を適切に表現可能と期待できる．

本研究では，オドメトリの系統誤差の軽減方法を提案する．系統誤差は同様

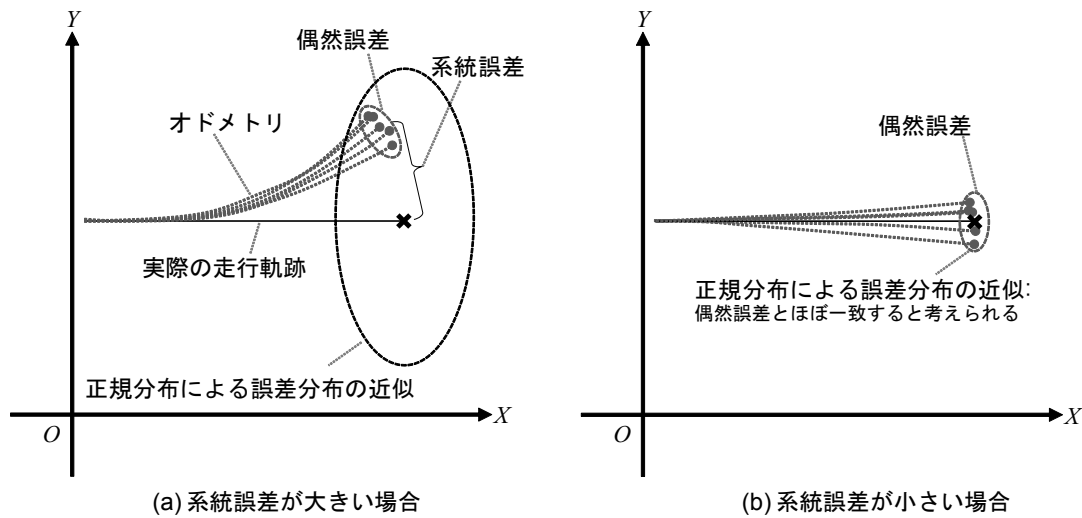


図 2.18: オドメトリの誤差と正規分布

に走行すると、同様の誤差が生じる。そこで、事前に走行した際のセンサの計測より、その環境で生じるオドメトリの誤差を得る。この事前に走行した際に生じた誤差より、自律走行時の系統誤差を推定し、修正する方法を提案する。

## 第3章 高所特徴のスキャンマッチングによる自己位置推定

一般的に、移動ロボットの自律ナビゲーションにおける自己位置推定では、直前の自己位置が既知であるとする、自己位置の追跡の方式が用いられる。この時、自己位置推定を破綻させないように誤差の増大を防ぐことが求められる。人の生活環境では、自己位置の推定誤差の増大における主な問題は、絶対位置推定の誤推定と考えられる。また、ランドマークの不足による絶対位置推定の不可能な状況で走行を続けると誤差が増大することも問題となる。本章では、これらの問題に対する地図上の絶対位置推定の改善として、計測を工夫する方法について述べる。

ランドマークの不足のため絶対位置推定が不可能となると、デッドレコニングの誤差を修正できないため、誤差が増大してしまう。この問題の原因として、歩行者や駐車自動車などによりランドマークの観測を妨げられることがあげられる。絶対位置推定の誤推定に対しては、不安定な計測による偽のランドマークが原因となる。生活環境では、歩行者などの時間による変化や、センサの計測がバラつき、不安定となる。計測な不安定は、地図と自律走行時の計測において一致しないため、偽のランドマークとなることがあり、これにより誤った位置を推定してしまう。

これらの主な原因は、歩行者などの人自身と駐車自動車などの人の活動による変化と考えられる。このため、人の活動による変化は人の背より高い場所においては少ないと考えられるため、高所では偽のランドマークが生じることが少ないと期待できる。また、人の頭越しに高所を計測することで、人にランドマークの観測を妨げられることを回避できると考えられる。

そこで，本研究では高所特徴のスキャンマッチングによる自己位置推定を提案する．スキャンマッチングは，スキャンデータ全体の形状特徴を用いる地図上の絶対位置推定の手法であり，ある程度計測できる対象があれば絶対位置推定が可能となる．ランドマークの観測の妨げ，および不安定な計測に対しては，水平にたいして上方に傾けて取り付けしたレーザスキャナにより高所を計測することで解決を図る．

本章では，初めに本研究で用いた ICP スキャンマッチングと，生活環境における絶対位置推定の課題である不安定な計測による影響について説明する．次に2次元レーザスキャナによる高所の計測方法について述べ，最後に実際の市街地での自律ナビゲーションの実験について述べる．

### 3.1 関連研究

路面に対して高い箇所は変化が少ないため，自己位置推定において有効な特徴が得られる．従来研究として，天井の画像から得られる光源の情報を用いて自己位置を推定する手法が提案されている [9][25][26]．しかし，これらの手法は自己位置推定に天井の画像を必要とするため，屋外では適用できない．本研究ではレーザスキャナの高所の計測を用いて自己位置を推定する手法を提案する．提案手法では，上方に傾けた2次元レーザスキャナを用いて高所を計測する．この高所の計測を，多様な生活環境における自己位置推定を可能とするため，スキャンマッチングに適用する，

本研究と同様にレーザスキャナによる高所の計測を用いた自己位置推定として，3次元レーザスキャナを用いた手法がある [17][41]．これらは，揺動機構に2次元レーザスキャナを搭載することで，高速に3次元の計測をしている．しかし，機構が複雑となるため，導入が難しくなり，メンテナンスが大変となることが課題としてあげられる．一方，本研究では2次元レーザスキャナを傾けることで適用可能であり，容易に適用可能な点が利点となる．



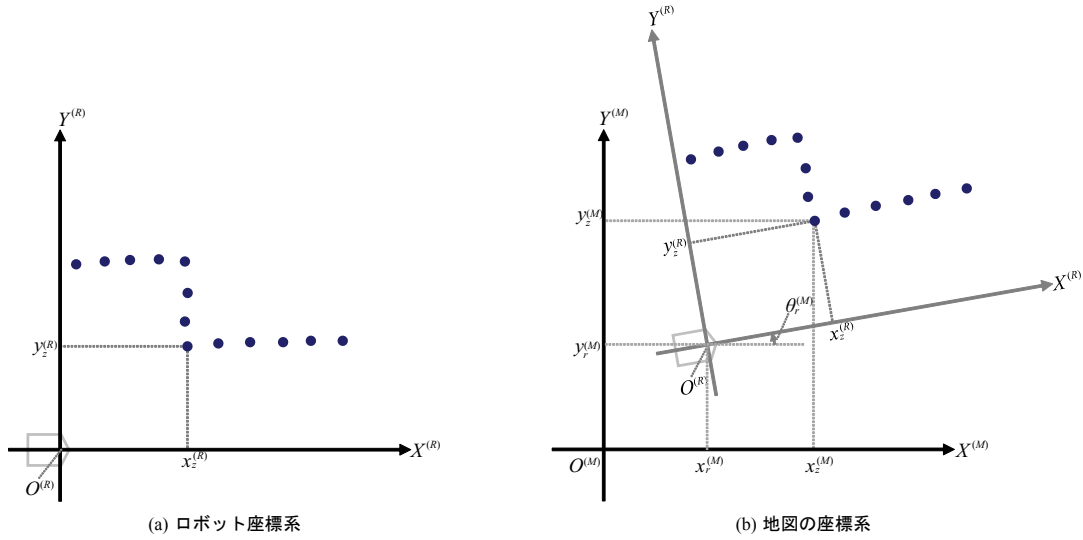


図 3.1: スキャンマッチングにおける座標系

## 3.2 スキャンマッチングとその生活環境における課題

ここでは本研究が用いた ICP(Iterative Closest Point) スキャンマッチングの中核となる ICP アルゴリズムの説明と生活環境における問題として環境の変化による影響について具体的に説明する。

### 3.2.1 ICP アルゴリズム

ICP アルゴリズムでは参照となる形状特徴に対して、入力となる形状特徴が一致する位置、姿勢を決定する。ICP スキャンマッチングはレーザスキャナのスキャンデータに ICP アルゴリズムを適用してロボットの位置、姿勢を推定する手法である。ロボットに搭載したレーザスキャナから得られた現時のスキャンデータ(現時スキャン)を入力とし、これと参照スキャンが一致するロボットの位置、姿勢を求める。参照スキャンには地図や少し前の時刻のスキャンデータが用いられるが、ここでは、地図を参照スキャンとして地図上の自己位置を推定する。形状特徴は点群もしくは面で表現され、点对点 [19] の対応付けや点对面 [20] の対応付けがある。ここでは、本研究で用いた点对点の対応づける方法について述べる。また、点对点の対応付けでは地図は点群で表現される。

ICP アルゴリズムの手順について説明する前に、スキャンマッチングにおける座標系について述べる。ロボットに搭載したレーザスキャナから得られるスキャンデータの各点は、ロボットの位置と姿勢により原点と軸を定義した座標系上の位置として得られる。厳密にはセンサの位置と姿勢により定義される座標系上の位置であるが、これはロボット上のセンサの位置、姿勢から、ロボットの座標系上の値に変換可能であり、ここでは簡単のため所与とする。図 3.1-(a) に示すように、ロボットの座標系上における  $i$  番目のスキャンデータの位置を  $\mathbf{z}_i = (x_i^{(R)}, y_i^{(R)})^T$  と示す。ここで、ロボットに搭載したセンサからは、直接、地図の座標系上の位置は計測できない。このため、地図と現時スキャンを対応付けるには、現時スキャンの点群  $Z = \{\mathbf{z}_i\}$  を、地図の座標系における位置に変換しなければならない。地図とロボットの座標系の関係は、図 3.1-(b) に示すように地図上のロボットの位置と姿勢、すなわち、自己位置  $\mathbf{x} = (x_r, y_r, \theta_r)^T$  によって定義される。本論文では、座標系の関係を親子と表現し、親の座標系上における原点の位置と軸の向きによって定義される座標系を子とする。ロボットの座標系は、親となる地図の座標系に対して、自己位置  $\mathbf{x}$  で定義される子の座標系になる。ここで、地図の座標系における現時スキャンの位置  $\mathbf{z}_i^{(M)} = (x_i^{(M)}, y_i^{(M)})^T$  は以下となる。

$$\mathbf{z}_i^{(M)} = T(\mathbf{x}, \mathbf{z}_i) \quad (3.1)$$

$$= \begin{pmatrix} \cos(\theta_r) & -\sin(\theta_r) \\ \sin(\theta_r) & \cos(\theta_r) \end{pmatrix} \mathbf{z}_i^{(R)} + \begin{pmatrix} x_r \\ y_r \end{pmatrix} \quad (3.2)$$

$T(\mathbf{x}, \mathbf{z}_i)$  は、 $\mathbf{x}$  で定義された子の座標系から親の座標系への位置  $\mathbf{z}_i$  の変換を示す。ここで、 $\mathbf{z}_i$  はセンサの計測により得られる定数であり、スキャンマッチングでは、ロボットの位置、姿勢  $\mathbf{x}$  を変数として、現時スキャンと地図の一致度が高くなる座標系を求めることになる。

ICP アルゴリズムの手順は大きく分けて以下の2つのステップとなる。

1. 地図とスキャンの点群の対応付け
2. 対応付けられた点間の距離が最小となる様に位置を推定

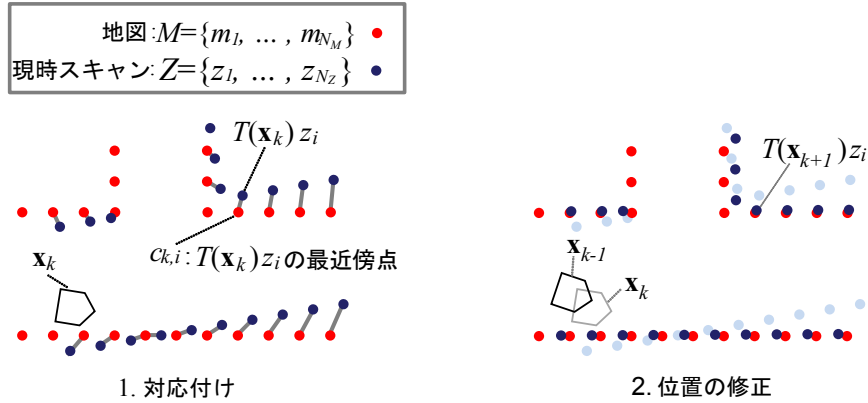


図 3.2: ICP アルゴリズム

この2つのステップを位置の推定結果が収束するまで反復して繰り返す．収束は，1 ループにおける位置の修正が閾値より小さくなるなどの条件で判断する．

現時スキャンと地図が一致する時，これらの2つの点群は重なる，厳密には計測の誤差などのため完全に重なることはないが，現時スキャンと地図と点群の最近傍の点との距離が0に近くなると考えられる．そこで，1の対応付けのステップ(図3.2-1)では，現時スキャンに対して地図の点群中で最近傍の点を対応付ける． $k$  回目の反復における，ロボットの位置を  $\mathbf{x}_k$  とするとき， $i$  番目の現時スキャンの点  $\mathbf{z}_i$  の対応点  $\mathbf{c}_{k,i}$  は以下となる．

$$\mathbf{c}_{k,i} = \underset{m_j}{\operatorname{argmin}} \|T(\mathbf{x}_k, \mathbf{z}_i) - \mathbf{m}_j\|^2 \quad (3.3)$$

ここで， $\mathbf{m}_j$  は地図の  $j$  番目の点を示す． $\mathbf{x}_0$  は ICP アルゴリズムにおける初期値であり，デッドレコニングによる推定値などを用いる．対応点との距離が十分近い時，地図と現時スキャンが一致していると考えられるため，一致度の評価関数は以下となる．

$$Score_{icp}(\mathbf{x}, Z, M) = - \sum_i \|T(\mathbf{x}, \mathbf{z}_i) - \mathbf{c}_{k,i}\|^2 \quad (3.4)$$

ここで， $Z = \{\mathbf{z}_i\}$  は現時スキャンの点群， $M = \{\mathbf{m}_i\}$  は地図の点群を示す．

次に2のステップでは，対応点間の距離が最小となるロボットの位置，姿勢を探索する．図3.2-2に示すように，対応点間の距離の総和がより小さくなる，すなわち，式3.4がより大きくなるロボットの位置，姿勢を，次のループのロ

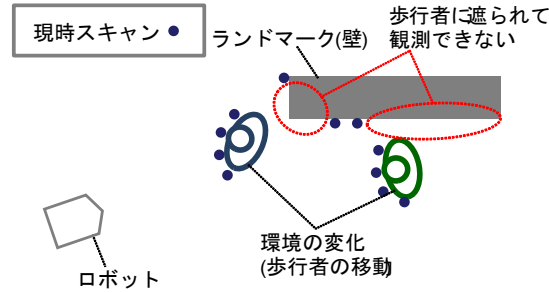


図 3.3: 環境の変化によるランドマークの観測の妨げ

ボットの位置，姿勢  $\mathbf{x}_{k+1}$  とする．具体的には，最適化手法を用いて次式を解く．

$$\mathbf{x}_{k+1} = \underset{\mathbf{x}}{\operatorname{argmax}} \operatorname{Score}_{icp}(\mathbf{x}, Z, M) \quad (3.5)$$

しかし，一般的には，評価値が最大にならずともある程度高くなった時点でステップ2を終了する．これは，ロボットの位置，姿勢が変化すると，現地スキャンに対する地図の最近傍点が変わることが考えられることから，ステップ1に戻って，再度対応付けをするためである．本研究では再帰的最適化手法であるニュートン法 [42] により，ロボットの位置，姿勢の更新を1度だけ行い，ステップ2を終了することとした．

### 3.2.2 環境の変化による影響

環境の変化による ICP スキャンマッチングへの影響として，以下の2つがあげられる．

- ランドマークの観測の妨げ

地図の作成時には観測できていた特徴が，環境が変化することにより観測できなくなることがある．図 3.3 に示す例のように，歩行者にセンサの視界を遮られるなどしてランドマークが観測できないことがある．このためランドマークの観測の機会が少なくなる問題がある．

- 偽のランドマーク

図 3.4 に示す例では，(a) センサによる観測においてランドマークではな

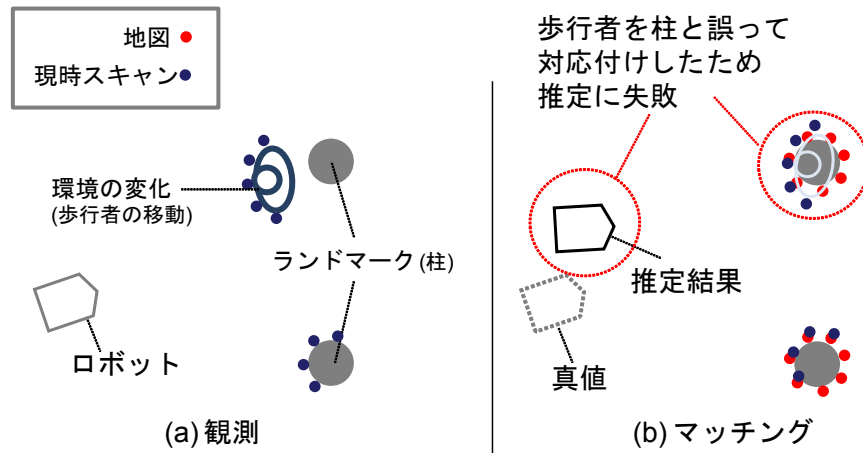


図 3.4: 環境の変化と地図の誤対応付け

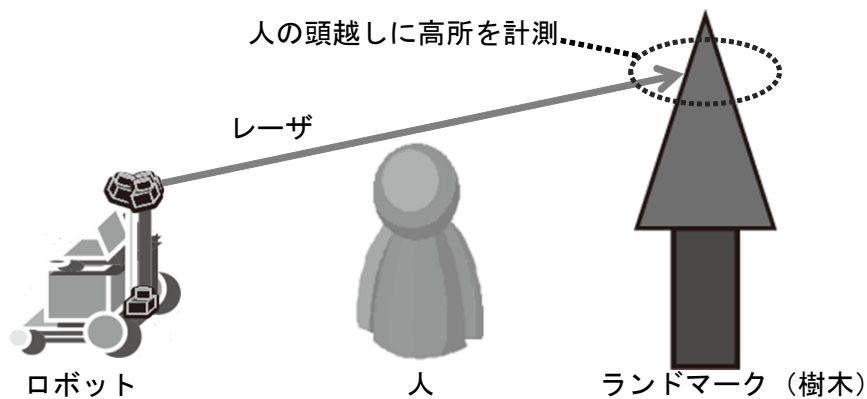


図 3.5: 2次元レーザスキャナによる高所の計測方法

い歩行者を観測している．(b)の自己位置推定においては，歩行者より得られた形状特徴を地図の柱と誤って対応付けたため，自己位置推定に失敗している．このように環境の変化が偽のランドマークとなり，地図の特徴と誤った対応付けた結果，絶対位置推定に失敗することがある．

本研究では，計測方法を工夫して高所の特徴を用いることで，これらの問題の解決を図る

### 3.3 高所の計測方法

生活環境における変化は主に人の活動によるものであり，人の頭上より高所では環境の変化が少ないと考えられる．そこで，環境の変化に影響を受けにくく

い絶対位置推定のために，高所の特徴を計測して，これを現時スキャンとして ICP スキャンマッチングにより自己位置を推定する．

図 3.5 に 2 次元レーザスキャナによる高所の計測方法について示す．傾けて取り付けたレーザスキャナにより人の頭上を計測する．これにより，歩行者や駐車自動車の移動などの人の活動による変化の少ない特徴を計測できる．

ここで，傾けて取り付けた 2 次元レーザスキャナでは，水平な平面ではなく立体的な空間が計測できる．しかし，本研究では実空間を水平な平面に近似した 2 次元の地図上での自己位置を推定する．そこで，任意の高さから 0.1m 程度の狭い範囲の特徴を用いて 2 次元のスキャンマッチングを行うことにした．樹木の枝葉のような立体的に複雑な形状物の特徴は平面上に近似できないため，0.1m 程度の狭い範囲のみを平面に近似し，2 次元のスキャンマッチングに用いることにした．

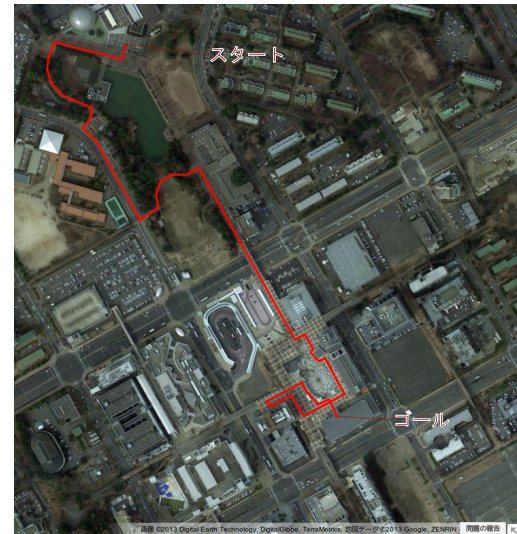
しかし，狭い範囲のみの特徴を用いる場合，1 スキャンからでは十分な特徴が得られず，自己位置の推定結果に誤差が生じたり，失敗することがある．そこで，1 スキャン，すなわち，1 箇所からのスキャンデータでは少ないデータ補うために，オドメトリによる自己位置推定結果をもとに，移動している間に計測した過去数回のスキャンデータを重ね合わせて用いることにした．ロボットの筐体次第ではあるが，一般的にオドメトリは数 m 程度の短い区間を走行する間であればその誤差は小さい．そこで，この短い区間を走行する間の過去数回のスキャンデータを，その時刻のオドメトリをもとに現在のロボットの座標系へ変換し，重ね合わせる．この重ね合わせたスキャンデータの現時スキャンとしてスキャンマッチングをする．

### 3.4 実験

高所特徴によるスキャンマッチングを用いて，実際の生活環境において自律走行実験を行った．実験はつくばチャレンジ 2009，2010 の課題のコースで行った．つくばチャレンジ 2009，2010 のコースを図 3.6 に示す．つくばチャレンジ 2009 は公園内を周回するコースであり，コース中には遊歩道，林間部，車道に



(a) つくばチャレンジ 2009



(b) つくばチャレンジ 2010

図 3.6: 実験環境

面した歩道，広場などがある．つくばチャレンジ 2010 では公園を抜け，つくば駅付近の市街地歩道を走行するコースである．つくばチャレンジでは，2009 では9日間，2010 では計8日間の実験走行および公式に記録をとる本走行の機会が設けられており，この間に提案手法を用いて自律走行実験を行った．以降，実験に用いたロボットのハードウェアと高所特徴によるスキャンマッチングを用いた自己位置推定と動作計画について説明したのち，走行結果について述べる．

### 3.4.1 ハードウェア

使用したロボットの外観を図 3.7 に示す．ロボットは独立二輪駆動型の移動機構であり，大きさは幅 0.75m，長さ 1.0m，高さ 1.5m である．ロボットの上部に北陽電機社製のレーザスキャナ，UTM-30LX を3台取り付け付けた．これらのレーザスキャナを用いて高所の特徴を計測する．レーザスキャナの取り付け位置について図 3.8 に示す．3つのレーザスキャナはそれぞれ右前方 ( $-45^\circ$ )，左前方 ( $45^\circ$ )，後ろ ( $180^\circ$ ) に向けて取り付け付けた．これらのレーザスキャナは 1.4m の高さの位置で上方へ  $10^\circ$  傾けている．これは，移動しているロボットから 2m 以内に人が近づくことは少ないと考えて，レーザスキャナの正面方向 2m の位





図 3.7: ロボットの外観

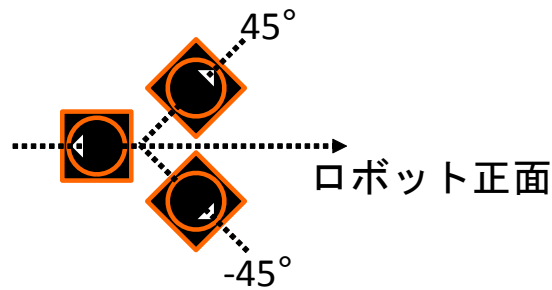


図 3.8: 上部レーザスキャナの取り付け

置にいる身長 1.7m の人の頭越しに計測できる様に取り付けた．これにより，環境の変化によりランドマークの観測を妨げられる問題の解決を図った．

### 3.4.2 高所特徴によるスキャンマッチングを用いた自己位置推定

本実験における自己位置推定の構成図を図 3.9 に示す．デッドレコニングにはオドメトリを用い，エンコーダのカウント値をもとにロボットの速度，角速度を推定する．自己位置の融合には拡張カルマンフィルタ用い，オドメトリよ



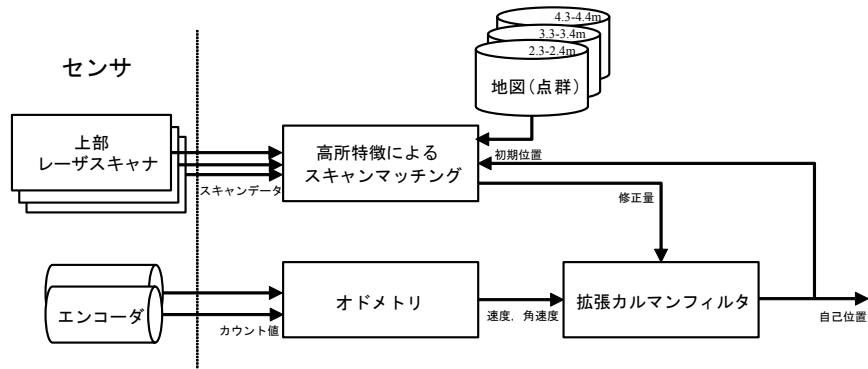


図 3.9: 高所特徴によるスキャンマッチングを用いた自己位置推定

りえられる速度，角速度を制御動作として，逐次自己位置およびその確率密度を推定する．高所特徴によるスキャンマッチングでは，ICP アルゴリズムの初期位置として，オドメトリにより更新されている現在の自己位置を用いる．ロボットの上部に取り付けたレーザスキャナにより計測した，高所の特徴を用いて ICP スキャンマッチングにより自己位置を推定する．この推定結果をもとに拡張カルマンフィルタにより，自己位置を修正する．ここで，拡張カルマンフィルタにより自己位置を推定するにあたって，高所特徴によるスキャンマッチングによる自己位置推定結果の確率密度を示す，誤差の共分散が必要となる．これは誤差の共分散は経験的に決めた固定値を用いた．スキャンマッチングの地図は事前にロボットをリモコンで操縦してコースを走行した際のセンサデータをもとに作成した．リモコン操縦時のオドメトリから得られる自己位置をもとに各自己位置でのスキャンデータの点群の位置を求める．各自己位置におけるスキャンデータを重ね合わせて地図を作成する．スキャンマッチングは 3 秒ごとに実行し，自己位置の誤差を修正することにした．ただし，スキャンマッチングによる推定結果が初期位置から大きく外れている場合は，推定に失敗したとしてこの結果は無視することにした．具体的には初期位置から距離が 1m 以上はなれている，もしくは向きが  $15^\circ$  以上ずれている場合は失敗と判断した．

スキャンマッチングに用いる特徴の高さについて人の身長は高くても 2.0m 程度でありこれより 0.3m 以上高い環境では人による環境の操作も少ないと考え，2.3m 以上の高さを用いることにした．しかし，特定の 1 つの高さのみでは環境

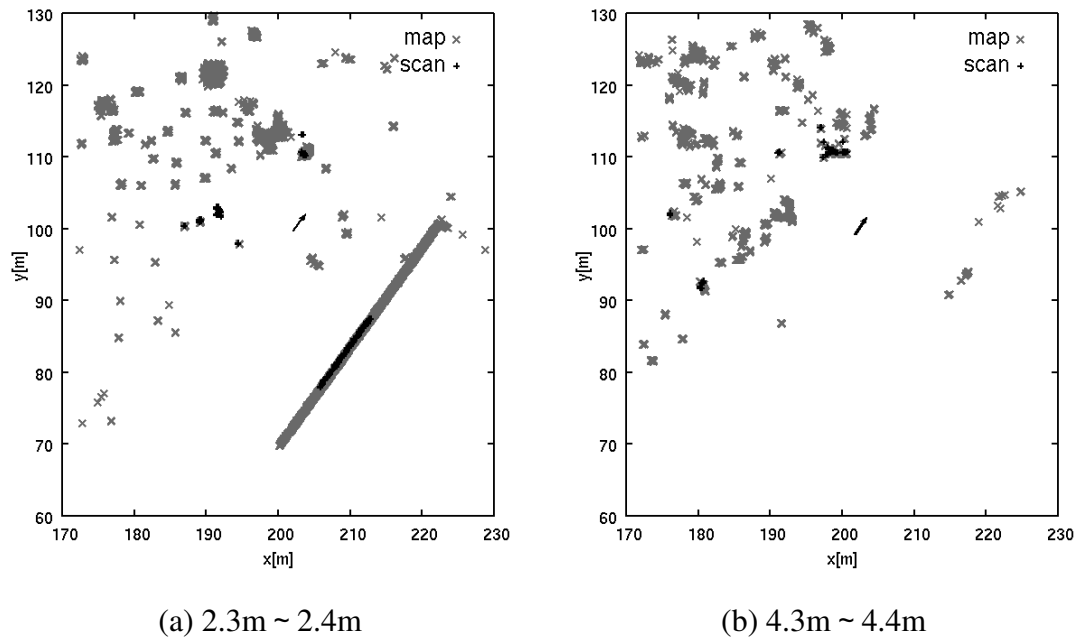


図 3.10: 高さによるスキャンデータの違い

次第で有意な特徴が得られないことが考えられるため、複数の高さの特徴を用いることとした。図 3.10 にある環境における 2.3m ~ 2.4m, 4.3m ~ 4.4m の高さのスキャンデータを示す。濃い灰色の+は上述の方法で計測した高所の特徴であり、薄い灰色の×は地図の点群である。この環境では、図 3.10-(a) に示す 2.3m ~ 2.4m の高さの範囲では図の右下にある直線のような特徴が計測できたが、一方、図 3.10-(b) に示す 4.3m ~ 4.4m ではわずかな特徴しか得られていない。高さの範囲は 2.3m ~ 2.4m, 3.3m ~ 3.4m, 4.3m ~ 4.4m を用意し、ある高さで絶対位置の推定に失敗した場合、別の高さでスキャンマッチングを行うことにした。

### 3.5 動作計画

ロボットはあらかじめ用意された経路地図をもとに現在位置から目的地まで走行する動作を計画する。図 3.11 に経路地図の概念を示す。経路地図はいくつかの点(サブゴール)とそれを結ぶ線分で表現し、ロボットはサブゴールまでの走行を順次計画、実行することで目的地まで走行する。また、経路地図には安全に走行可能な領域として、各サブゴール間の線分に対して道幅を与えている。ロボットはこの領域内での走行を計画する。この与えられた領域内で走行を計

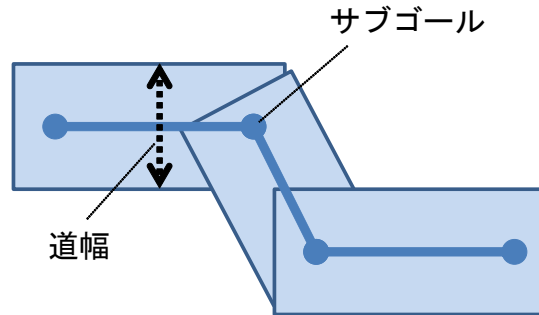


図 3.11: 経路地図

画するためには、ロボットは地図上の自己位置を常に知る必要がある。一方で、このように安全に走行可能な領域をあらかじめ与えておくことで、路面の舗装や縁石、路肩などの静的な環境の認識は基本的に不要となり、歩行者や自転車搭乗者などの移動障害物のみ認識すればよいため、環境認識の問題が容易となる。

ロボットは基本的にサブゴール間を結ぶ線分上を走行し、水平に取り付けたレーザスキャナにより障害物が検出された場合、回避を計画する。回避の計画についてのイメージを図 3.12 に示す。障害物を黒い領域で、サブゴール間の線分を灰色の矢印で示している。この例ではサブゴール間の線分上に障害物があり、走行が不可能である。ここで、空間を格子状に区切る。道幅より走行可能とされた領域(薄い灰色の領域)の上に、障害物の位置とロボットの大きさをもとに走行不可能な領域(濃い灰色の領域)を重ね合わせる。走行が可能として残った領域の格子内でダイクストラ法により障害物を回避する経路を求め、この経路上の走行を計画する。

### 3.5.1 走行結果

上述の自己位置推定手法を用いて自律走行実験を行った結果、つくばチャレンジ 2009、2010 とともに本走行において完走に成功し、実験走行の日程を含めて 2009 では計 11 回、2010 では 12 回の完走に成功した。

図 3.13 につくばチャレンジ 2010 コース (1.1km 強) における高所特徴による

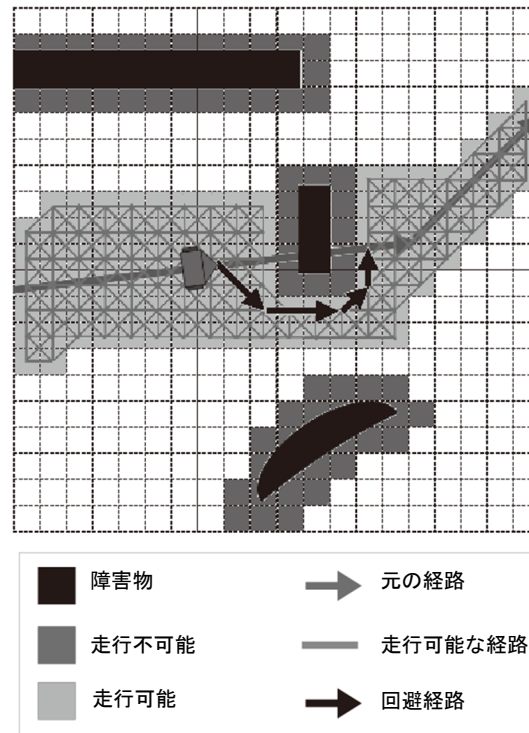


図 3.12: 障害物回避

スキャンマッチングにより自己位置を修正したポイントを赤い点で示す．高所特徴による自己位置推定が 777 箇所成功している．コース全体を通して高所特徴によるスキャンマッチングで自己位置修正が可能であり，10m 程度のわずかな区間を除いて自己位置の修正に成功している．高所特徴を用いたスキャンマッチングができなかった 10m の区間は図 3.14 に示すように高所に計測できる特徴がなかったことがわかる．

高所特徴によるスキャンマッチングにより環境地図の座標系上において合計， $x$  軸方向に -3.3m， $y$  軸に -0.8m，姿勢を 54 度，修正している．1.1km の長距離コースであるにもかかわらず位置 ( $x$ ， $y$  軸方向) の修正が少ない．一方，姿勢は大きく修正している．これはオドメトリによる誤差が大きくなる前にロボットの姿勢が正確に推定できたため，位置のずれが少なかったと考えられる．結果として，幅 1.2m 程度の狭い区間も障害物回避をすることなくスムーズに通過することができた．これは予定した軌跡上を正しく走行できたためであり，自己位置の推定精度の高さを表している．

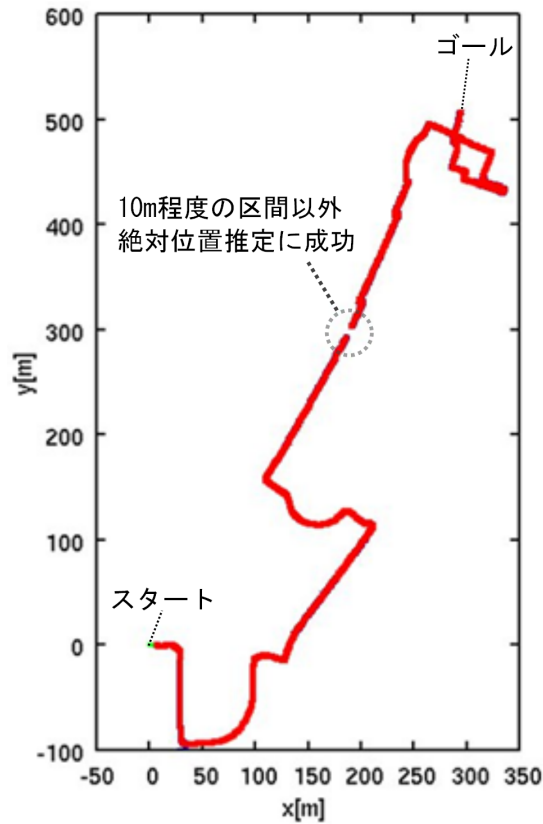


図 3.13: つくばチャレンジ 2010 における高所特徴によるスキャンマッチングに成功した箇所

### 3.6 成果

人の生活環境における自己位置推定として、高所特徴のスキャンマッチングによる自己位置推定を提案した。また、提案手法を用いた自己位置推定の実装例を示し、実際の生活環境における実験について述べた。

生活環境における自己位置推定における問題であるとして、ランドマークの不足と絶対位置推定の誤推定があげられる。高所を人の頭越しに計測することで、ランドマークの不足の問題に対して、歩行者等によるランドマークの観測の妨げられることを回避する。同時に、絶対位置推定の誤推定に対しては、人の活動による変化の少ない高所の特徴を用いて自己位置を推定することで、絶対位置推定の誤推定の原因である不安定な計測による偽のランドマークが生じることを防ぐ。また、本手法を用いた自己位置推定による自律ナビゲーション



図 3.14: 高所特徴を用いたスキャンマッチングに失敗した区間

により、つくばチャレンジ 2009(1.0km 強)のコースで 11 回、つくばチャレンジ 2010(1.1km 強)のコースで 12 回の走行に成功している。また、つくばチャレンジ 2009、2010 とともに本走行において完走に成功している。つくばチャレンジの本走行において完走したチームは 2009 では 64 チーム中 5 チーム、2010 では 64 チーム中 7 チームのみであり、本手法の有用性の高さを示している。また、遊歩道、林間部、車道に面した歩道、広場、市街地歩道など様々な環境が存在するこれらのコース中において、本手法が適用出来なかった区間は 10m 程度のみであり、本手法の実用性の高さを示している。

提案手法の課題としては、高所に特徴がない環境があげられる。実験を行った環境では 10m のみの区間であったが、周囲に背の高い物体の無い区間があり、このような区間では自己位置が推定できない。これについては、他の絶対位置推定手法を併用するなどの対策が必要となる。また、搭載するセンサやその取り付けに制約があることも課題としてがあげられる。実験における実装では、高所に 3 つのレーザスキャナを取り付けた。代わりの方法としては 3 次元レーザスキャナや 2 次元レーザスキャナに無限回転機構や揺動機構を取り付けること

が考えられるが、3次元レーザスキャナは高価であり、無限回転機構や揺動機構は機構が複雑でありメンテナンスが大変になる。

## 第4章 統計量に基づくレーザスキャナ の尤度場を用いた自己位置 推定

本章では，人の生活環境における絶対位置推定について，不安定な計測による誤推定の問題を解決する方法として，レーザスキャナの統計量に基づく尤度場を用いた自己位置推定について述べる．

自己位置推定では，現在の自己位置を見失わないように誤差の増大を防ぐことが重要となる．人の生活環境では，自己位置の推定誤差が増大する原因として絶対位置推定の誤推定があげられる．絶対位置推定の誤推定は，歩行者などの移動や，ロボットの振動などの計測条件の変化により，計測がバラつき，不安定となることが原因となる．この不安定な計測に対して，第3章では，不安定な計測が少ないと考えられる高所特徴を用いて自己位置を推定する方法を提案した．この手法では，市街地や公園など様々な生活環境において有効に働く絶対位置推定を実現した．しかし，高所に物体のない区間では適用できないことと，高所を計測するためセンサの取り付けに制約があることが課題としてあげられた．そこで，計測方法に依存しない方法として，不安定な計測にロバストな絶対位置推定の手法について検討した．本研究では，計測方法に依存しない方法として，レーザスキャナの取り付け方として一般的な方法であるロボットに水平に取り付けた場合を想定して，不安定な計測にロバストな絶対位置推定の手法を考案する．

ロボットの位置，姿勢が十分近い場合，センサの計測は同様となることが期待される．そこで，絶対位置推定では事前に取得したセンサの計測をもとに地



図を作成し，この地図と自律走行時の計測が一致する位置，姿勢を自己位置として推定する．しかし，生活環境では環境が激しく変化するためセンサの計測が不安定となり，ロボットの位置，姿勢が十分近い場合でもセンサの計測が異なる値を示すことがある．このため，地図作成のためにセンサの計測を取得した際に生じた不安定な計測の特徴が，地図に記載されると誤推定の原因となる．

ここで，誤推定の原因となる不安定な計測は計測の度に異なる値となることがあり，バラつく．一方で，頻繁に同様の計測が得られる対象は自己位置推定にとって有意といえる．レーザスキャナでは複数回計測した際に，不安定な計測における反射点の位置はバラつく，一方頻繁に反射点が観測される位置の特徴が有意と考えられる．本研究では，各位値における反射点観測の尤度を用いて地図とスキャンデータの一致度を評価する．この時，尤度は事前に収集した複数回の計測から得られる統計量をもとに計算する．この尤度により地図に自己位置推定に有意な特徴を強く，不安定な計測を弱く重み付けて，一致度を評価することができる．この一致度の評価を用いて自己位置を推定することで，不安定な計測にロバストな自己位置推定が可能と期待される．

本章では，初めに関連研究について述べる．次に，生活環境におけるレーザスキャナの不安定な計測とその主な原因を述べる．次に，提案手法である尤度の計算方法について述べ，この尤度を用いて地図とスキャンデータとの一致度の評価方法を説明する．また，提案手法の有効性を示すために，実際の生活環境で計測されたスキャンデータを用いて，他の手法との一致度の評価を比較した結果を示す．最後に，提案手法をパーティクルフィルタにおけるパーティクルの重み付けに適用して自己位置推定し，実際に人が活動する環境における自律ナビゲーションの実験を行った結果について述べる．

## 4.1 関連研究

絶対位置推定の地図は記述する環境の情報によって2つにわけられる．1つは静止物や特定の物体など環境を構成する要素を記述した地図であり，もう1つは環境で得られるセンサの計測について記述した地図である．前者では，間

取り図等の人間の地図を用いることが考えられるが、これらは走行する環境に対して必要な地図が用意されているとは限らない。このため、一般的には事前に走行した際のセンサの計測から環境を認識して作成する。レーザスキャナを用いた絶対位置推定では、占有格子地図 [43][44] が広く用いられる。後者は事前に走行した際のセンサの計測値をそのまま地図としたり、特徴を抽出して作成する。例としては、ICP スキャンマッチングなどに用いる反射点を点群で記述した地図 [45][46] や、反射点の分布を記述した地図 [47] がある。

環境を構成する要素を記述した地図を作成する場合、生活環境では不安定な計測のため環境の認識が困難となる。詳細は4.2節にて述べるが、生活環境(特に人の身長より低い高さ)では様々な原因で計測が不安定となる。不安定な計測を引き起こす原因すべてに対応することは難しく、地図を適切に表現できないことがある。レーザスキャナを用いた手法では占有格子地図を用いた自己位置推定 [13][17][48] が用いられるが、生活環境ではロボットの振動や傾きにより、計測上で静止物の位置が曖昧となる問題ある。[13][17][48] ではこの問題に対しては解決しておらず、[13] では、オドメトリの誤差を考慮し、誤推定を検出することで自己位置推定の破綻を回避している。[17] では、3次元レーザスキャナを用いて不変な箇所の計測を多く得ることで対処している。[48] では、磁器センサの計測を用いて姿勢の誤差を修正することで、誤推定が生じないようにデッドレコニングの誤差を軽減している。

環境で得られるセンサの計測について記述した地図を作成する場合、計測のバラつきを表現する方法が必要となる。これに対して、反射点観測の尤度を用いる方法がある。反射点観測の尤度とは、反射点がある位置で観測されたとき、それが環境中で観測されたとして尤もらしいかを示す値である。この尤度を用いると、計測のバラつきは尤度の低さとして表現できる。各位置に対する尤度を記述した地図は尤度場と呼ばれる。[18] ではビームによる測距センサの計測モデルを考慮し、静止物の位置とセンサの計測誤差から尤度場を求める方法について述べている。しかし、この方法では静止物の位置を知るために、占有格子地図や間取り図を必要とする。そこで、本研究では事前に収集した複数回のスキャンデータの統計量をもとに尤度場を作成する方法を提案する。この方法

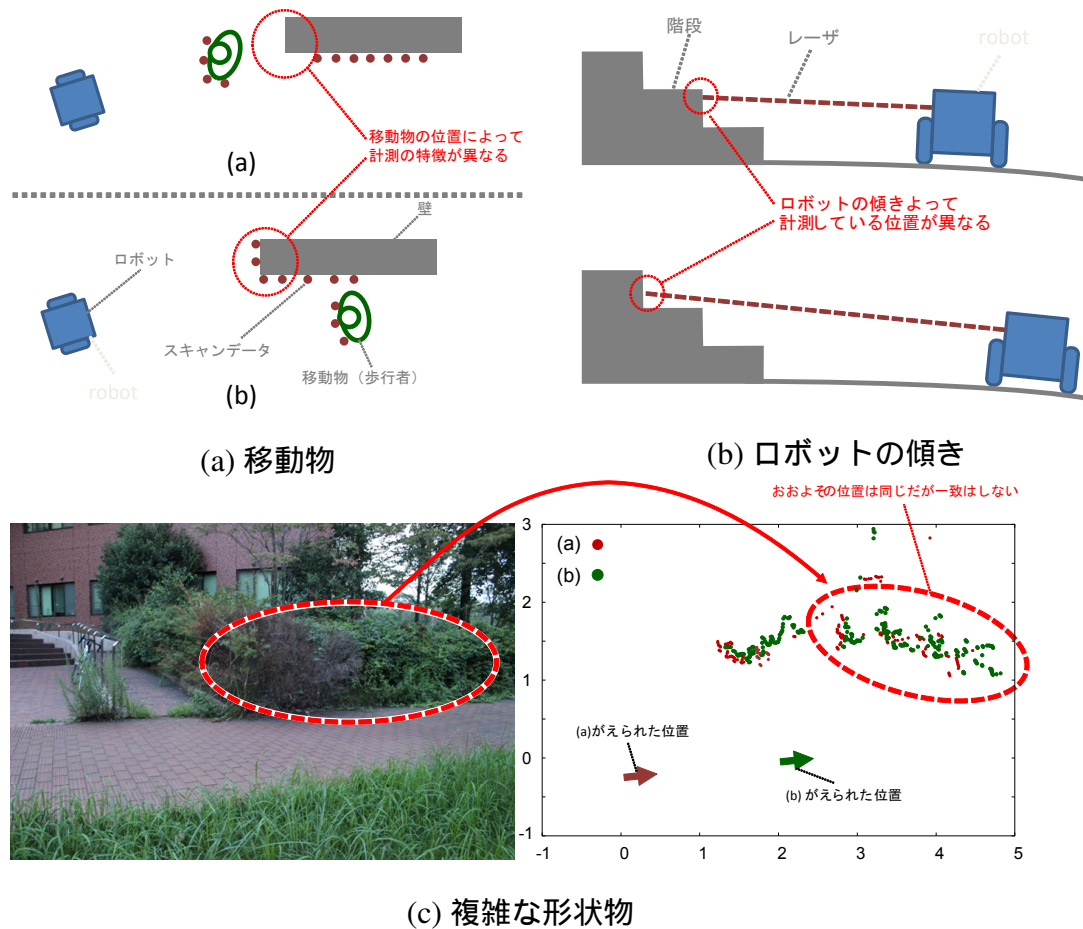


図 4.1: 生活環境における不安定なスキャンデータ

では，静止物の位置や不安定な計測となる原因に関わらず，収集した複数回のスキャンデータの点群をもとに尤度を計算する．本研究では，作成した尤度場を用いてスキャンデータの一致度を評価することで，不安定な計測にロバストな絶対位置推定を実現する．

## 4.2 生活環境における不安定なスキャンデータ

ここでは，生活環境においてスキャンデータが不安定となる原因について主な例をあげる．

- 歩行者などの物体の移動

図 4.1-(a) に示すように歩行者などの移動物は，時間により位置が変化するため，同様のスキャンデータが得られるとは限らない．また，物体の移

動による不安定な計測は動的な移動物の他に，駐車自動車の移動などの一時的な静止物の移動によっても生じる．

- ロボットの傾きや振動による計測位置のずれ

生活環境，特に屋外では路面に傾斜や凹凸があるため，ロボットの筐体が振動したり，傾いたりする．この時，図 4.1-(b) に示すように，ロボットに搭載されたレーザスキャナで計測している位置が大きく変化することがある．このため，階段や地面の隆起などを計測した場合，その計測値が異なってしまうことがある．筆者の経験上，ロボットに水平にレーザスキャナを取り付けた場合に，計測位置のずれによる不安定な計測は絶対位置推定において特に問題となる．生活環境の人の身長より低い位置では，ベンチ，階段，花壇のプランター，地面の隆起など様々な形状物がある．これらは，ロボットの傾きや振動のため計測する位置の変化し，見えたり，見えなかったり，あるいは測距値バラつく．このため，これらの対象は静止物であるにも関わらず，センサの計測上ではその位置は曖昧であり，地図上で適切に表現することが難しい．さらに，生活環境では，多数のベンチやプランターなどが並べられていたり，階段や地面の隆起が大きくセンサに写ることがしばしばあり，絶対位置推定が困難となることがある．

- 樹木の枝葉やフェンスなど複雑な形状物 樹木の枝葉やフェンスの様な複雑な形状物はレーザが隙間を透過ことがあるため，その奥を計測することがあり，スキャンデータが安定しない．図 4.1-(c) は同じ樹木の枝葉を別の位置で計測した結果を示しているが，赤と緑の2つのスキャンデータが重ならず，一致しないことが確認できる．

### 4.3 統計量に基づくレーザスキャナの尤度場

尤度場を用いた自己位置推定について図 4.2 にイメージ図を示す．(a) に示すように，ある区間で事前に収集した複数回のスキャンデータを重ね合わせた時，反射点が密に観測された箇所(壁面)と，不安定な計測(移動している歩行者)の

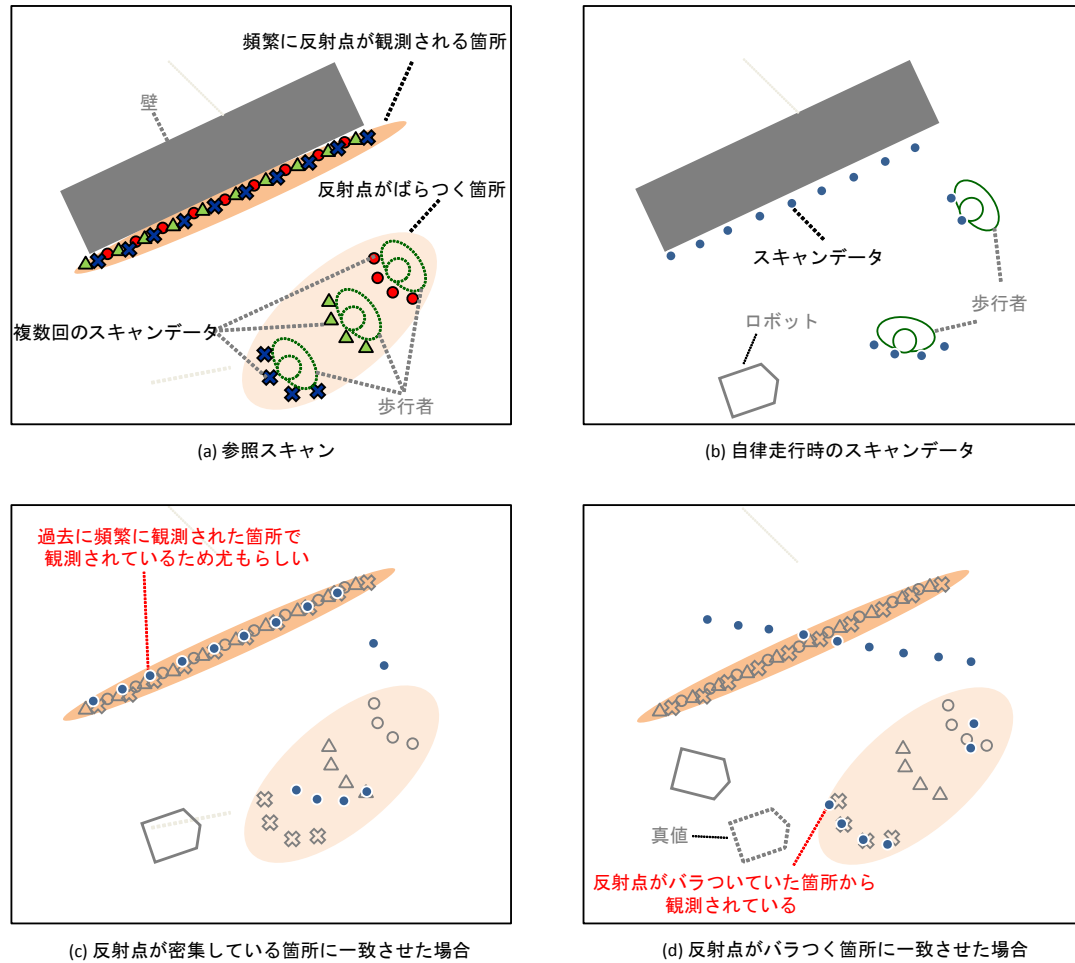


図 4.2: 尤度場とスキャンデータの一致度の評価

ため反射点が疎に観測された箇所があるとする．この区間でもう1度計測した際には，反射点が密に観測された箇所から頻繁に反射点が観測され，疎に観測された箇所ではあまり観測されないと考えられる．ここで，(a)の計測とは別に(b)に示すような計測が得られたとする．この時，(c)に示すのように，事前に収集したスキャンデータ中の反射点が密に観測された箇所から，多くの反射点を得られたとするロボットの位置，姿勢は尤もらしく，そうではない(d)のロボットの位置，姿勢は誤りであると推測できる．このように，事前に複数回のスキャンデータの反射点から，あるロボットの位置，姿勢を評価できる．

以降，尤度場の作成のために事前に複数回のスキャンデータから尤度の計算方法について述べる．次に，この方法により作成した尤度場とスキャンデータ

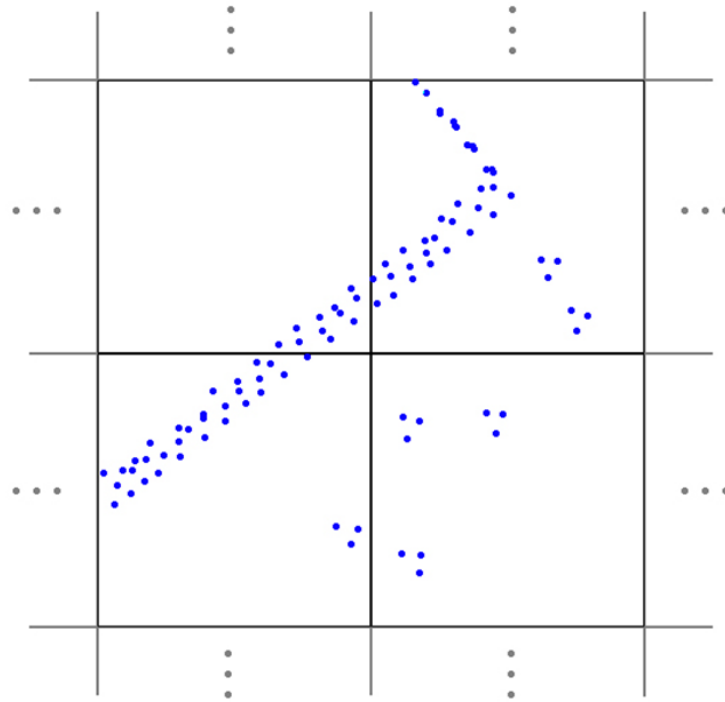


図 4.3: 格子状の地図とそのセル

の一致度を評価する方法について述べる．

### 4.3.1 尤度の計算

事前に複数回の計測したスキャンデータをもとに尤度を計算する．この事前に収集したスキャンデータを参照スキャンとする．参照スキャンより尤度の計算に必要な統計量  $m$  を集計する．図 4.3 に示すように地図の空間を均等な格子状に区切り，各セルにおける以下を集計する．

- セル内の点の数:  $n$
- セル内の点の位置の平均値:  $\mu$
- セル内の点の位置の共分散行列:  $\Sigma$

ここで，地図の空間をセルに区切り離散化する際に，セルのサイズが大きい場合，粒度が荒くなる問題がある．粒度が荒い場合，特徴の位置を明確に表現

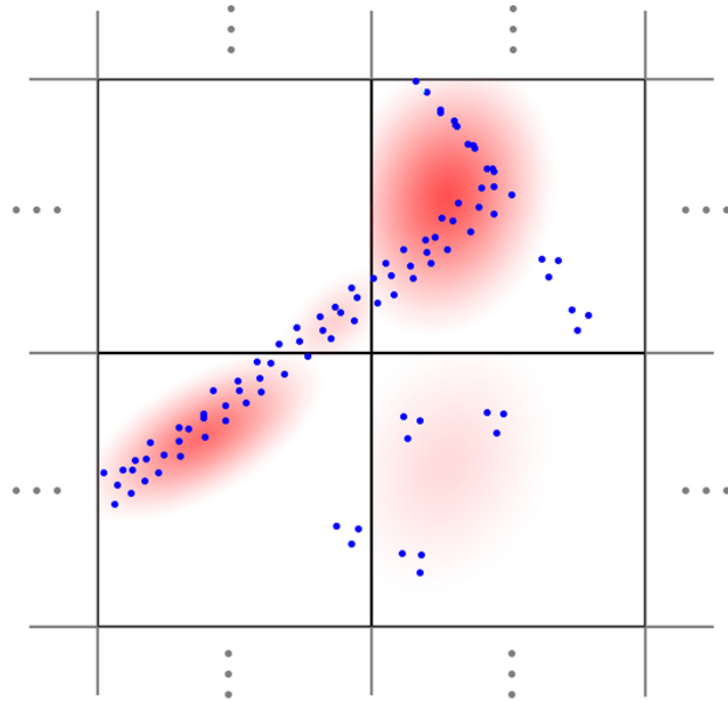


図 4.4: 統計量に基づく反射点観測の尤度

できず、自己位置推定の精度が低下する。一方、セルのサイズが小さい場合、信頼できる統計量を得るには大量のスキャンデータが必要となる。そこで、本手法では NDT[47] を参考に大きめのセルで区切ることとした。NDT はスキャンデータの全体の形状特徴をランドマークとする絶対位置推定手法である。NDT では、提案手法と同様に地図の空間を格子状に区切り、地図の形状特徴を各セル内における点群の正規分布を用いて表現する。この正規分布と現時スキャンの一致度を評価することで、自己位置を推定する。この時、NDT における地図は  $1m \times 1m$  程度の大きさのセルを用いている。本研究では、NDT の地図と同様に  $1m \times 1m$  程度の大きめのセルで区切り、各セル内における点群の正規分布を用いる。ただし、本研究では各セル内における点群の正規分布を各セル内における反射点観測の確率密度として用いる。

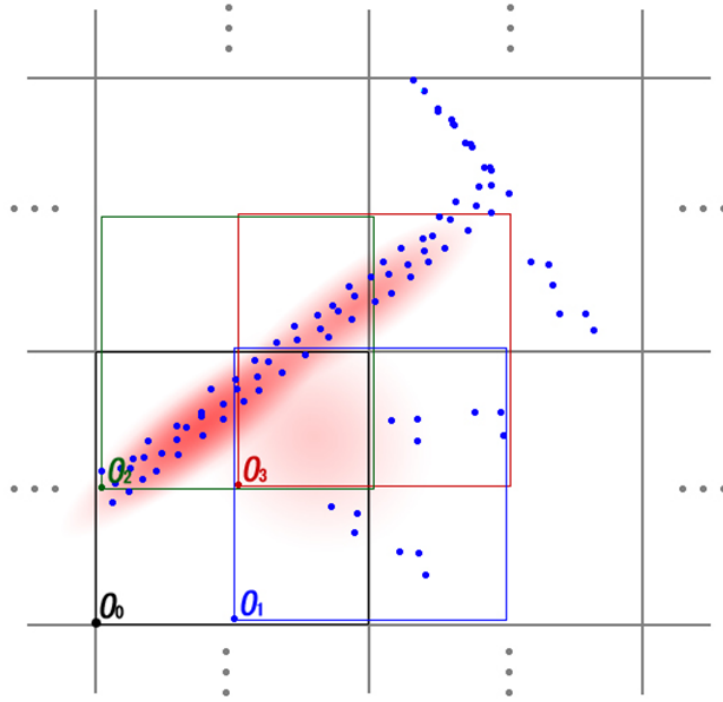


図 4.5: 4 種類のセルの分割

ある位置  $\mathbf{a} = (x_a, y_a)^T$  を示す反射点が観測されたとき，これが参照スキャンの統計量  $m$  を得た環境で観測されたとする尤度  $l(\mathbf{a}|m)$  は以下で計算される．

$$l(\mathbf{a}|m) = \frac{n_a}{N} \cdot \eta_a \frac{1}{2\pi \sqrt{|\Sigma_a|}} \cdot \exp\left(-\frac{(\mathbf{a} - \mu_a)^T \Sigma_a^{-1} (\mathbf{a} - \mu_a)}{2}\right) \quad (4.1)$$

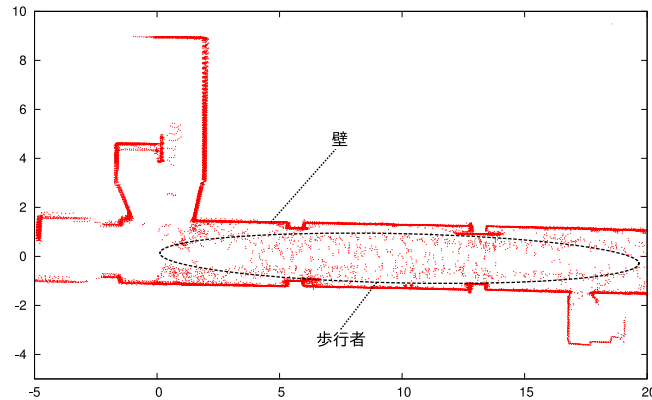
本手法では，尤度をセル内で反射点が観測される頻度と，セル内の反射点の分布をもとに計算する． $n_a$ ， $\mu_a$ ， $\Sigma_a$  は  $\mathbf{a}$  の位置を包含するセルにおける各統計量であり， $N$  は参照スキャン全体における反射点の数を示す． $\eta_a$  は正規化係数であり，以下によって求められる．

$$\eta_a = \left( \iint_{S_a} \frac{1}{2\pi \sqrt{|\Sigma_a|}} \cdot \exp\left(-\frac{((x, y)^T - \mu_a)^T \Sigma_a^{-1} ((x, y)^T - \mu_a)}{2}\right) dS_a \right)^{-1} \quad (4.2)$$

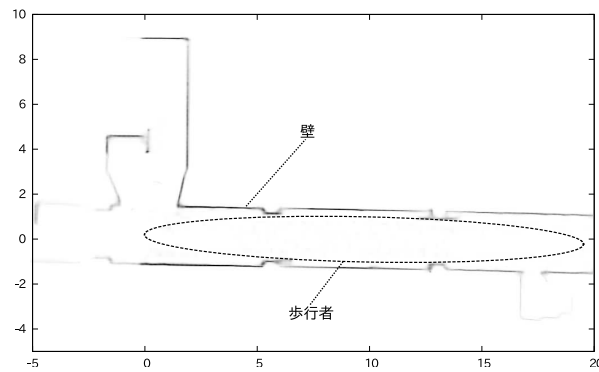
$S_a$  は  $\mathbf{a}$  の位置を包含するセルの範囲を示す．式 4.1 による尤度のイメージ図を図 4.4 に示す．図 4.4 では，尤度を赤の濃淡で示す．この赤い領域はセル内の点群の正規分布に従って広がり，セル内の点の数に応じて濃淡を表現している．

しかし，地図の空間を格子状に区切るため，離散化の影響によりセルの境界付近では適切に尤度が計算できない．そこで，図 4.5 中の黒，赤，緑，青の枠で





(a) スキャンデータの点群



(b) 尤度場

図 4.6: 提案手法による尤度場：歩行者のいる廊下環境における尤度場

示すように，セルの境界を横軸，縦軸それぞれセルの大きさの半分ずつずらし  
 て，4 つそれぞれで統計量  $M = \{m_0, m_1, m_2, m_3\}$  を集計する．この4 つそれぞれ  
 で計算される尤度を重ねることで，セルの境界の尤度を互いに補完できる．最  
 最終的にこれらの統計量による式 4.1 の平均値を尤度として用いる．

$$L(\mathbf{a}|M) = \frac{1}{4} \sum_{j=0}^3 l(\mathbf{a}|m_j) \quad (4.3)$$

図 4.6，4.8 に提案手法による尤度の計算を用いて作成した尤度場の実例を示  
 す．図 4.6 では，歩行者が移動している廊下環境における尤度場を示す．図 4.6-  
 (a)，参照スキャンとなる点群を赤い点で示す．この環境では周囲の壁が安定し  
 て計測できる．一方で，壁の内側には歩行者が移動により生じた不安定な計測  
 が確認できる．図 4.6-(b) は (a) の点群を参照スキャンとして作成した尤度場を

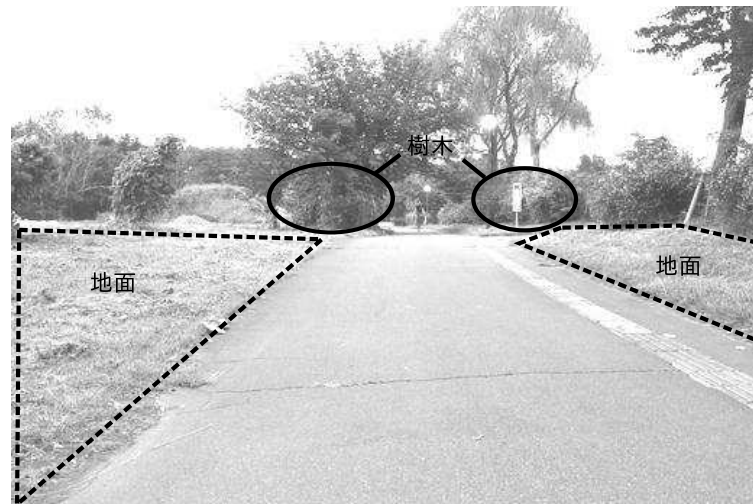


図 4.7: 提案手法に尤度場：ロボットの振動により地面のスキャンデータが不安定となる環境

示す．図 4.6-(b) は尤度場をビットマップで表現している．各ピクセルの位置に対して，式 4.3 による尤度を黒の濃淡で示しており，黒いほど尤度が高いことを示す．歩行者の特徴はほぼ消えていることに対して，壁の特徴は強く重み付けられている．この尤度場を用いて自己位置を推定することで，移動物のような不安定な計測の特徴に影響されず，自己位置が推定できると考えられる．

図 4.8 にロボットの傾きや振動により計測が不安定となる環境における尤度場の例，図 4.7 にその環境の写真を示す．この環境では，道の両側に地面の隆起がある．この地面の隆起はロボットの振動により計測される位置が変化するため，不安定に計測される．この環境で，ロボットが道なりに走行した際のスキャンデータを参照スキャンとして尤度場を作成した．参照スキャンを図 4.8-(a) に示す．スキャンデータの点群より，地面の隆起の点群が広い範囲に分布していることが確認できる．図 4.8-(b) では地面の隆起の特徴は，薄く広く示されている．これは地面の隆起の特徴は，計測値が曖昧であり，有意な特徴とならないことを表現している．一方で，樹木の特徴に尤度が高くなっていることがわかる．

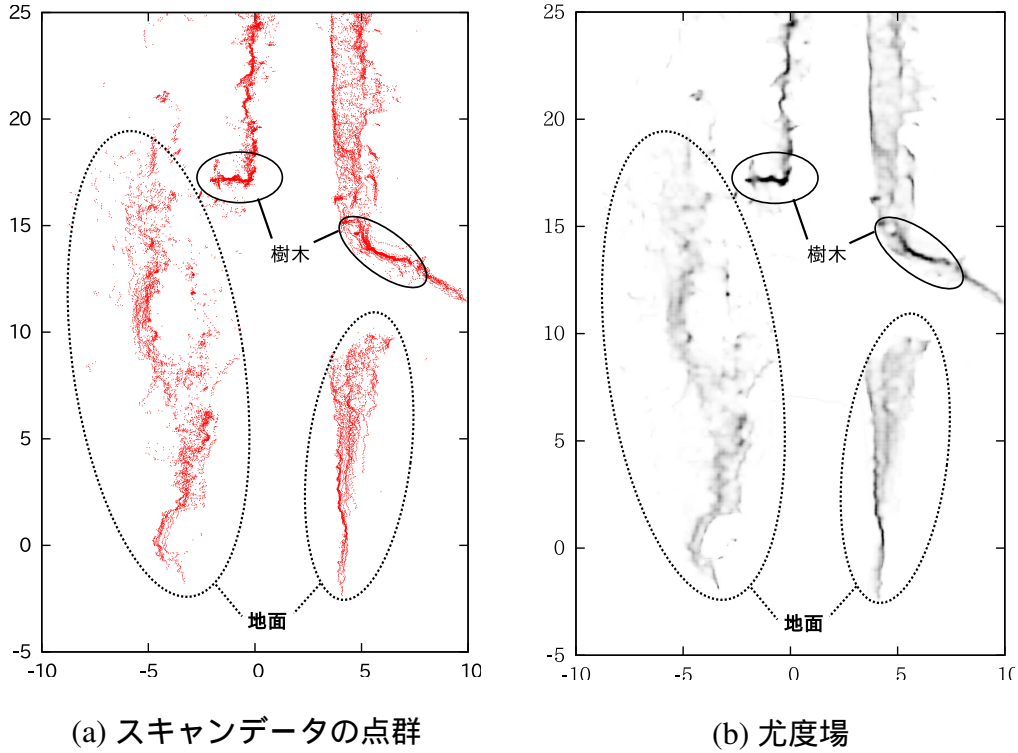


図 4.8: 提案手法による尤度場：ロボットの振動により地面のスキャンデータが不安定となる環境における尤度場

### 4.3.2 地図とスキャンデータとの一致度評価

一致度の評価は、現時スキャンの点の位置に対する尤度の総和とした。ロボットの座標系における現時スキャンの  $i$  番目の点の位置を  $\mathbf{z}_i = (x_i^{(R)}, y_i^{(R)})^T$  と示す。また、地図の座標系における  $i$  番目の点の位置  $\mathbf{z}_i^{(M)} = (x_i^{(M)}, y_i^{(M)})^T$  と示す。第3章で述べた ICP アルゴリズムと同様に、レーザスキャナから得られる現時スキャンはロボットの座標系上の位置  $\mathbf{z}_i$  であるが、一致度の評価においては地図の座標系上の位置  $\mathbf{z}_i^{(M)}$  が必要となる。これは、ロボットの位置、姿勢  $\mathbf{x} = (x, y, \theta)^T$  をもとに式 3.2 により求められる。ロボットの位置が  $\mathbf{x}$  であるとき、地図に記載された統計量  $M$  と現時のスキャンデータ  $Z = \{\mathbf{z}_i\}$  の一致度の評価値  $Score(\mathbf{x}, Z, M)$  を以下となる。

$$Score(\mathbf{x}, Z, M) \propto \sum_i L(\mathbf{z}_i^{(M)}) \quad (4.4)$$

### 4.3.3 地図とスキャンデータの一貫度評価の比較

ここでは、他の方法による地図とスキャンデータの一貫度の評価との比較について述べる。比較対象は、占有格子地図との照合による手法とNDTにおける一貫度の評価の2つとした。占有格子地図との照合は、不安定な計測の原因の1つである動的に変化に対して有効な方法として広く用いられる。NDTは不安定な計測に対して考慮された手法ではないが、本手法が参考としたため、これらと比較することとした。これらの手法における具体的な一貫度の評価方法は後述することとして、まず比較方法について説明する。

ここでは、ロボットの位置、姿勢に対する各手法による一貫度の評価値を比較する。ある位置で計測したスキャンデータを用いて一貫度を評価する。スキャンデータを得た位置を中心に $5m \times 5m$ の範囲に対して、縦軸、横軸 $0.05m$ 毎にロボットの位置を変化させ、スキャンデータと地図の一貫度を評価値を計算した。これは、パーティクルフィルタによる自己位置推定において、 $5m \times 5m$ の範囲に $0.05m$ 毎に配置された10000個のパーティクルに対して、スキャンデータをもとにパーティクルに重みを与える事に相当する。パーティクルフィルタにおいて、パーティクルの重みは計測モデルによる推定の確率分布となる。確率的自己位置推定において、絶対位置推定の誤推定とは、推定結果の確率分布が適切でない問題と考えられる。そこで、パーティクルの重みの分布を比較することで本手法の評価とした。パーティクルフィルタにおいては、本来のロボットの位置、姿勢の評価値が低いと、真値付近のパーティクルが淘汰され、自己位置推定が破綻することがある。また、誤った位置、姿勢の評価が高いと、パーティクルが広く分布し、やがて誤差が増大することがある。このため、真値付近の位置、姿勢の評価が高く、他の位置、姿勢では評価が低くなることが求められる。

評価値は図4.10-(a), (b), (c)に示すように青色の濃淡として表現した。評価方法が妥当であれば、ロボットの位置が真値に近いければ評価値は高く、遠いければ低くなることが期待されるため、評価した範囲の中心が濃く、中心から離れるほど薄く表示されていれば、正しく自己位置を推定しており、そうでな

ければ、誤推定の恐れがある．また、スキャンデータを、各手法において評価値が最大となったロボットの位置をもとに赤い点で示している．一致度の評価が適切であれば、地図の特徴とスキャンデータが重なることが期待される．

比較対象の手法による地図とスキャンデータの一致度の評価方法について述べる．初めに占有格子地図との一致度の評価について述べる．占有格子地図では、地図の空間を細かい格子状に区切り、各セルにその空間が物体に占有されている確率(占有度)を設定する．占有格子地図は環境に固定された静止障害物の有無を示した地図であり、この地図とのマッチングでは歩行者等、動的物体のスキャンデータの影響を受けにくい．このため、人の生活環境における自己位置推定のために用いられる．本実験では、占有格子地図とスキャンデータの一致度の評価は、占有空間で観測されたスキャンデータの点の数とし、ロボットの位置、姿勢  $\mathbf{x}$  の時の占有格子地図  $M_{occ}$  とスキャンデータ  $Z$  の評価関数  $Score_{occ}(\mathbf{x}, Z, M)$  を以下とした．

$$Score_{occ}(\mathbf{x}, Z, M_{occ}) \propto \sum_i \begin{cases} 1 & (p_i > 0.5) \\ 0 & (otherwise) \end{cases} \quad (4.5)$$

ここで、 $p_i$  は占有格子地図  $M_{occ}$  に記された  $\mathbf{z}_i^{(M)}$  の位置が占有度である．なお、セルのサイズを  $5cm \times 5cm$  として、地図を作成した．

NDT では格子状に区切ったセル内における参照スキャンの分布により、参照スキャンの形状特徴を表現する．提案手法はNDTを参考にしたため、NDTと類似する点が多いが、提案手法では環境の情報を参照スキャンから得られる尤度で表現していることに対して、NDTではセル内の点群の分布による形状で表現している点で本質的に異なる．NDTにおける地図とスキャンデータの一致度の評価は以下となる．

$$w_{ndt}(\mathbf{z}_i^{(M)}, M) = \exp\left(-\frac{(\mathbf{z}_i^{(M)} - \mu_i)^T \Sigma_i^{-1} (\mathbf{z}_i^{(M)} - \mu_i)}{2}\right) \quad (4.6)$$

$$W_{ndt}(\mathbf{z}_i^{(M)}, M) = \frac{1}{4} \sum_{j=0}^3 w_{ndt}(\mathbf{z}_i^{(M)}, m_j) \quad (4.7)$$

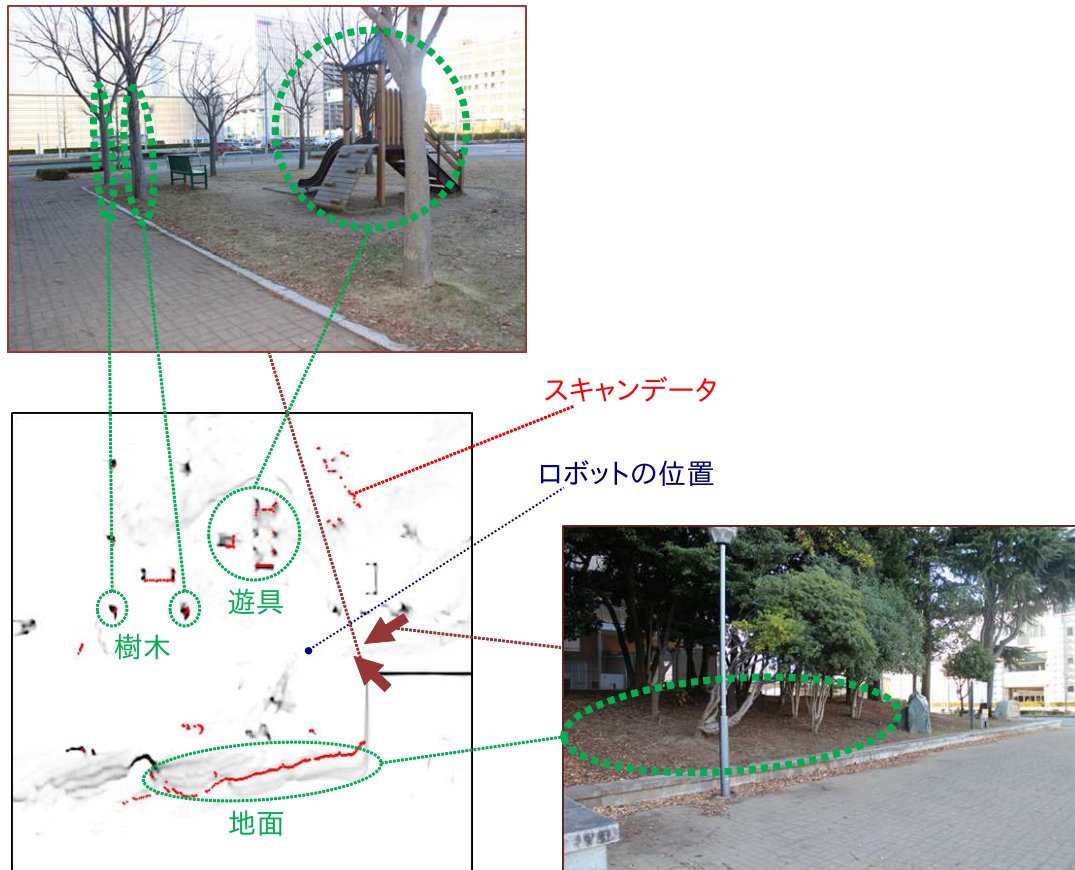


図 4.9: 地面のスキャンデータが不安定となる環境

$$Score_{ndt}(\mathbf{x}, \mathbf{Z}, M) \propto \sum_i W_{ndt}(\mathbf{z}_i^{(M)}, M) \quad (4.8)$$

なお，それぞれ手法による評価値を比較するため，10000 箇所の評価値の合計が1となるように正規化した．また，いずれの方法においても，地図の作成には同じ参照スキャンを用いた．参照スキャンは，ロボットを操作して走行しながら，北陽電機社製レーザスキャナ UTM-30LX を用いて収集した．ロボットが0.1m 走行する毎に，その時点の計測を参照データとして用いた．また，一致度の評価に用いたスキャンデータは参照スキャンの収集とは別の機会にロボットが走行した際に取得したスキャンデータを用いた．

不安定なスキャンデータとして，地面が計測される環境において一致度の評価を比較した．図 4.9 に環境の地図と写真を示す．図 4.9 では，一致度を評価したスキャンデータを赤い点で，スキャンデータを取得したロボットの位置を青

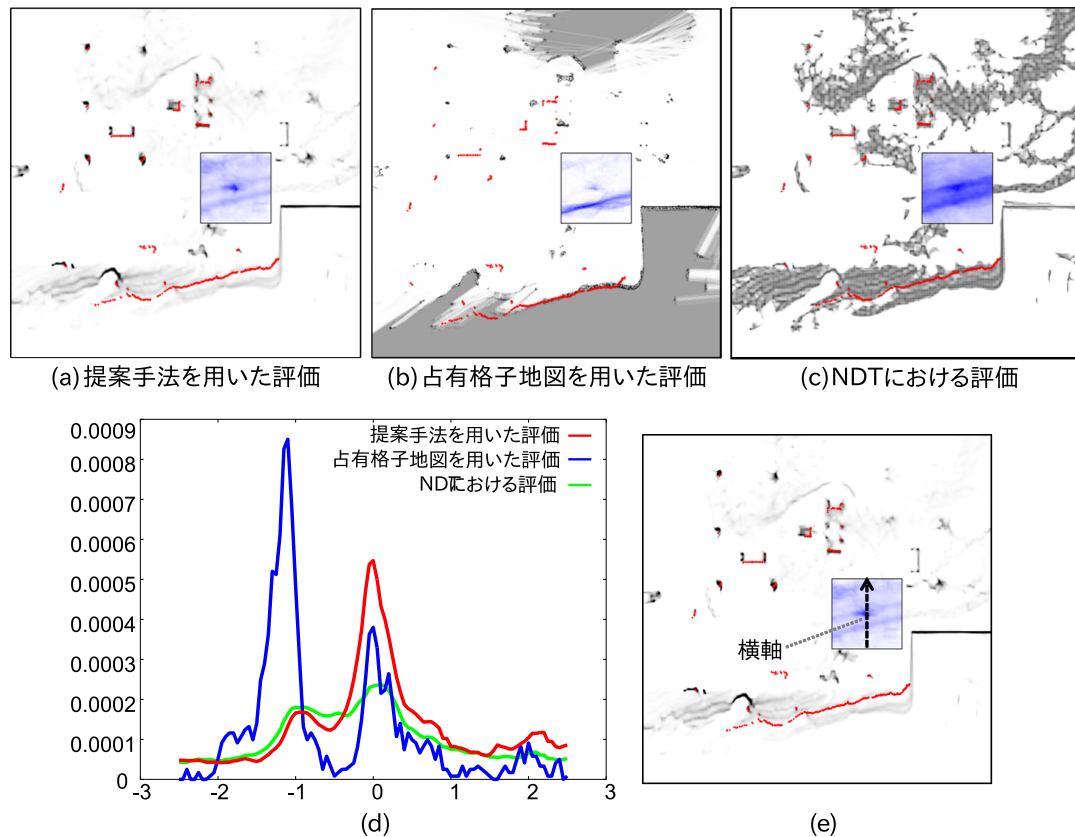


図 4.10: 一致度の評価値の比較:地面のスキャンデータが不安定となる環境

い丸，写真を撮影した位置を矢印で示す．この環境では，ロボットの振動のため地面の隆起が不安定に計測される．この地面の隆起が広い範囲で計測される一方で，樹木や遊具などわずかではあるが安定して計測される特徴がある．図 4.10 に各手法による一致度の評価値を示す．図 4.10-(a)，(b)，(c) に各方法による評価値を，(d) には (e) の矢印が示す方向に横軸をとり，各位置の評価値をグラフで示す．なお，各地図における黒の濃淡は以下を示す．

- 提案手法: 式 4.3 による尤度
- 占有格子地図: 静止物に占有されている確率
- NDT: 式 4.7 による評価値

提案手法では，赤い点で示されたスキャンデータが樹木や遊具の特徴と重なっている．また，青の濃淡で示した一致度の評価値は，スキャンデータを取得した



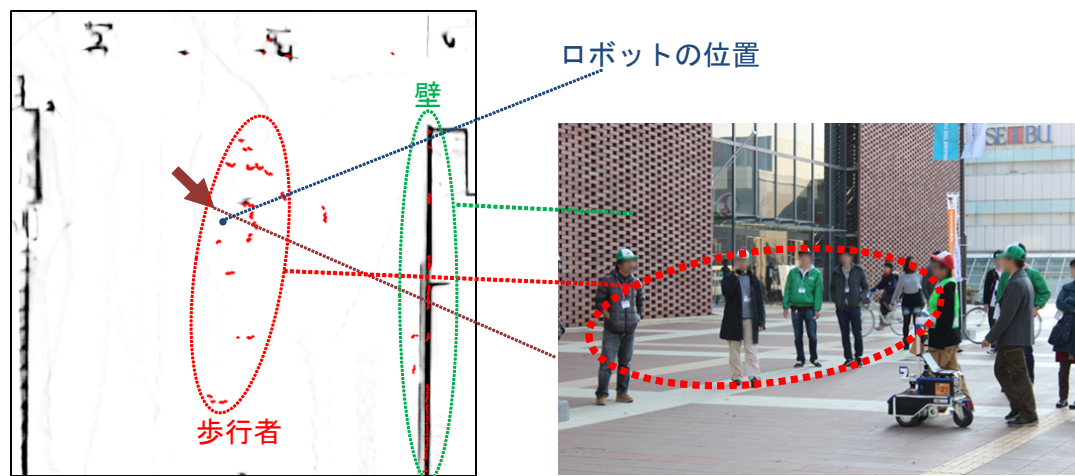


図 4.11: 人通りの多い環境

ロボットの位置付近のみ濃い色を示しているため、適切に自己位置を推定できることが確認できる占有格子地図を用いた一致度の評価では、地面の不安定なスキャンデータに対して、占有空間を適切に表現できない。このため、地図に記された地面のスキャンデータによる占有空間が偽のランドマークとなり、誤った位置における一致度の評価が高くなっている。また、(e) からスキャンデータをとった位置の付近より、誤った位置の評価値が高くなっていることが分かる。NDT における一致度の評価では、地図に示された形状特徴が明確ではない。これは不安定な計測に対して考慮した手法ではないため、不安定な計測により特徴を適切に表現できないことが原因と考えられる。結果として、濃い青が広く分布し、ロボットの位置が正しい位置から離れている場合でも、一致度の評価が高くなっている。

多数の歩行者がいる環境における一致度の評価について、図 4.11 に環境の地図と写真を示すこの環境では、建物の壁が安定して計測できるため、これをランドマークとして自己位置が推定できる。しかし、図 4.11 の写真に示すように、人通りが多いため、歩行者や自転車搭乗者などの計測が不安定となる。図 4.12 に各手法による一致度の評価値を示す。図 4.12 から、NDT では、参照スキャンを収集した際の歩行者の特徴がもやの様に地図に記載され、地図上の偽のランドマークとなっていることが確認できる。一方、提案手法と占有格子地図では



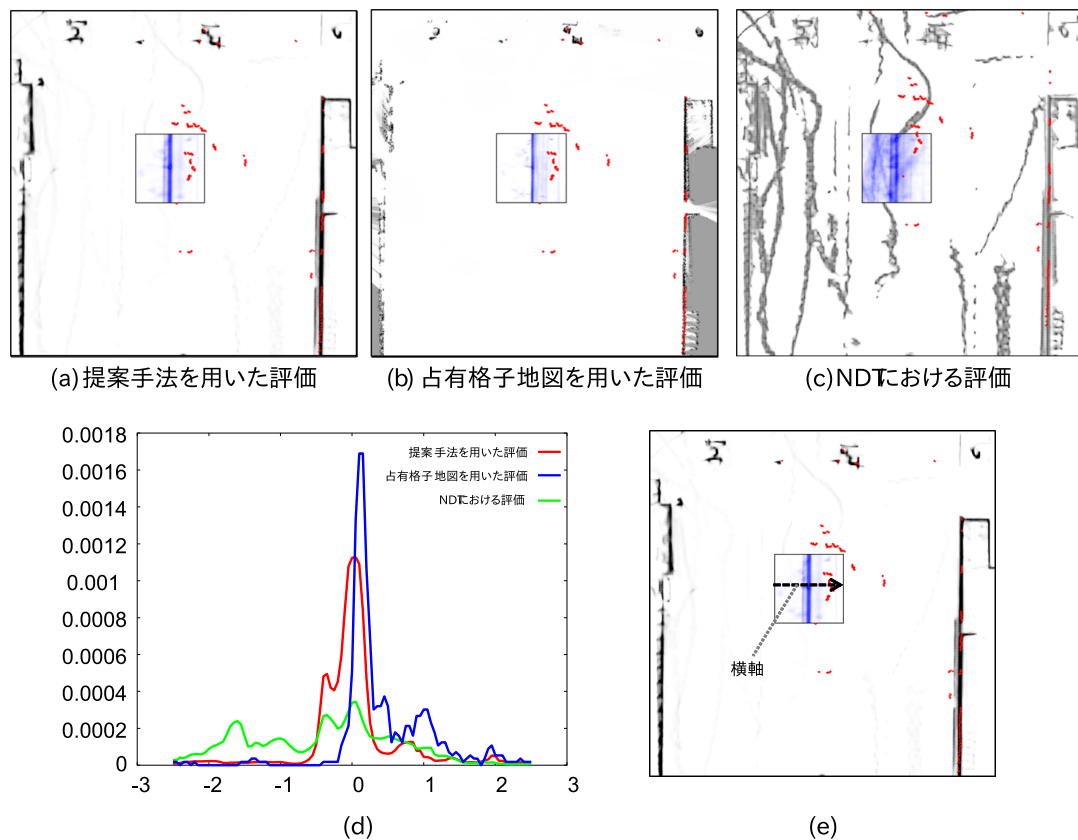


図 4.12: 一致度の評価値の比較:人通りの多いにおける一致度の評価

歩行者の特徴は記載されていない．占有格子地図は，静止物に占有されているか否かを特徴としており，動的に移動する歩行者に左右されず，適切に静止物を特徴を示している．このため，提案手法と占有格子地図ではランドマークとなる建物の壁の特徴が一致する位置の評価値が高く，適切にロボットの位置，姿勢を評価していることが確認できる．一方で，NDT では地図上の歩行者の特徴が一致し，様々な位置の評価値が高くなってしまっている．

以上より，提案手法はスキャンデータが不安定となる人の生活環境においても，適切に地図とスキャンデータの一致度を評価できることが確認できる．このことから，提案手法を用いた自己位置推定は，生活環境の不安定な計測に対して有効だと期待される．

## 4.4 自律走行実験

提案手法を用いて、実際の生活環境において有効に働くか確認するために、自律走行実験を行った。実験はつくばチャレンジ 2011 の課題のコースで行った。つくばチャレンジ 2011 のコースを図 4.13 に示す。つくばチャレンジ 2011 のコース中には遊歩道、林間部、広場、つくば駅付近の市街地歩道、屋内繁華街などがある。つくばチャレンジでは、2011 では計 8 日の実験走行および公式に記録をとる本走行の機会が設けられており、この間に提案手法を用いて自律走行実験を行った。以降、実験に用いたロボットのハードウェアと自己位置推定について説明したのち、走行結果について述べる。なお、動作計画では第 3 章と同様の手法を用いた。

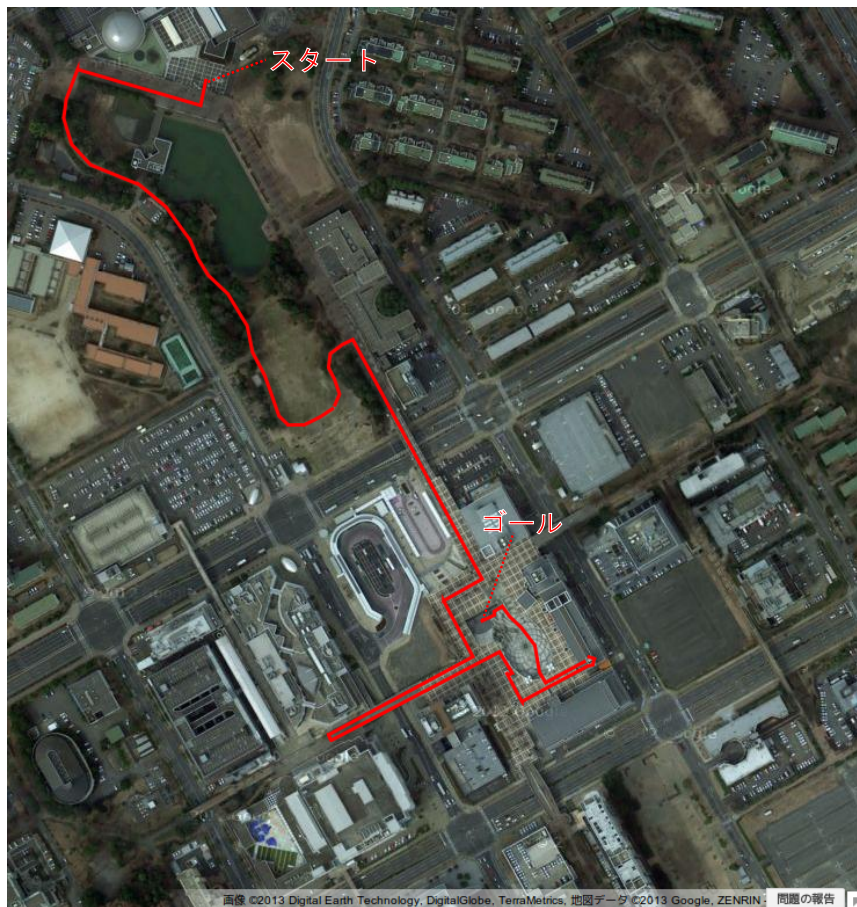


図 4.13: 実験環境:つくばチャレンジ 2011 の課題コース



図 4.14: ロボットの外観

#### 4.4.1 ハードウェア

使用したロボットの外観を図 4.14 に示す．ロボットは独立二輪駆動型の移動機構であり，大きさは幅 0.55m，長さ 0.80m，高さ 0.55m である．外界センサには北陽電機社製のレーザスキャナをロボットの前方に向けて，地面に対して水平に取り付けた．このレーザスキャナを自己位置推定に用いた．

#### 4.4.2 自己位置推定

本実験における自己位置推定の構成図を図 4.17 に示す．自己位置推定にはパーティクルフィルタを適用した．左右の車輪のエンコードの計測をもとにオドメトリに従って，パーティクルを遷移させる．水平に取り付けたレーザスキャナの計測値をもとに，提案手法によりこれらのパーティクルを重み付け，リサンプリングすることで，自己位置を推定する．

時刻  $t$  における  $i$  番目のパーティクルを  $\mathbf{x}_{t,i} = (x_{t,i}, y_{t,i}, \theta_{t,i})^T$ ，パーティクルの集合を  $\chi_t = \{\mathbf{x}_{t,i}\}$  と示す．動作モデルにはオドメトリを用いた．各パーティクル

は左右の車輪のエンコードのカウント値をもとに、オドメトリに従って移動する。この時、オドメトリに生じる誤差を反映させるため、各パーティクルの位置、姿勢に対して、タイヤの回転量に比例した分散をもつランダムな誤差  $\varepsilon_{t,i}$  を与える。

$$\mathbf{x}_{t,i} = \mathbf{x}_{t-1,i} + \Delta t \cdot \begin{pmatrix} v_t \cos(\theta_{t-1,i}) \\ v_t \sin(\theta_{t-1,i}) \\ \omega_t \end{pmatrix} + \varepsilon_{t,i} \quad (4.9)$$

ここで、 $\Delta t$  はエンコードのサンプリング周期であり、速度  $v_t$ 、角速度  $\omega_t$  は左右の車輪の回転量をもとに式 2.12 により計算する。

水平に取り付けたレーザスキャナの現時のスキャンデータ  $Z_t$  と提案手法による尤度場の一致度の評価値をパーティクルの重みとした。時刻  $t$  における  $i$  番目のパーティクルの重み  $w_{t,i}$  は以下となる。

$$w_{t,i} = \text{Score}(\mathbf{x}_{t,i}, Z_t, M) \quad (4.10)$$

この重みにより<sup>1</sup>、リサンプリングを行う。パーティクルの数は常に 500 個としてリサンプリングを行った。また、パーティクルの重み付けとリサンプリングは 0.2 秒ごとに行うこととした。ただし、ロボットが移動していない場合は、パーティクルの重み付けとリサンプリングは行わないこととした。ロボットの移動はオドメトリにより検知する。自己位置の推定結果にはパーティクルの重みによるすべてのパーティクルの位置、姿勢の重み付き平均を用いた。

地図の作成には、事前にロボットをリモコンで操作してコースを走行した際のスキャンデータを用いた。ロボットが 0.1cm 走行する毎に、その時点の計測を参照データとして用いた。

#### 4.4.3 走行結果

上述の自己位置推定手法を用いて自律走行実験を行った結果、つくばチャレンジ 2011 の本走行において完走に成功し、実験走行の日程を含めて計 5 回の完

<sup>1</sup>この実験の際には、式 4.2 を解くことができなかったため、正規化係数  $\eta_a = 1$  として実験を行った

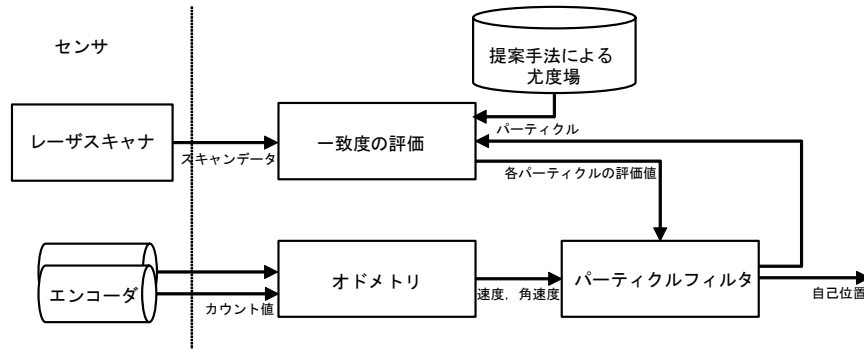


図 4.15: 尤度場とスキャンデータの一貫度評価を用いた自己位置推定

走に成功した。つくばチャレンジ 2011 では 66 チーム中、本走行での完走に成功したチームは 6 チームのみであり、本手法の有効性を示している。休日における実験走行では図 4.17 に示すような人混みに置いても、自己位置推定が破綻することなく目的地まで走行することに成功している。経路中には図 4.17 に示すような、幅 2m 程度の細いスロープを登る区間があり、この区間ではロボットの傾きにより計測が不安定になるが、この区間も安定して通過している。このように、遊歩道、林間部、広場、つくば駅付近の市街地歩道、屋内繁華街などの様々な環境において、水平に取り付けたレーザスキャナ 1 つで自己位置推定に成功しており、また、計測が不安定な区間でも高い精度で自己位置の推定に成功したことが確認できる。

## 4.5 検討

ここでは、本手法を実用する際の検討として、参照スキャンの収集について知見を述べる。本手法は参照スキャンとして収集した複数回のスキャンデータの統計量から得られる尤度をもとに、環境中の不変な箇所に重み付けすることで不安定な計測にロバストな自己位置推定を実現する。このとき、不変な箇所に重みづけるためには、多様な状況におけるスキャンデータを偏りなく収集することが求められる。

ここで、状況とは歩行者などの周囲の環境の状況とロボットの位置や傾きな





図 4.16: 人混みを走行する様子



図 4.17: 細いスロープを走行する様子

どのロボット自身の状況を意味する．周囲の環境の状況は操作できず，時間により変化する．ここで，時間による変化は様々である．歩行者の移動のように短期的な変化もあれば，駐車自動車の移動のように間欠的な変化もある．計測の間隔に比べて十分速い短期的な変化の場合，計測上では常に変化しているためデータの偏りの問題はない．一方，間欠的な変化はデータの収集時に変化しないことがあり，問題となる可能性がある．ロボット自身の状況としては，ロ

ボットの位置や傾きなどがある．これについては，走行予定の経路において等間隔でスキャンデータを収集すればよい．

そこで，本研究における実験では，ロボットが移動する毎にスキャンデータを収集した．具体的にはロボットが0.1mの距離を移動する度に収集した．この時，ロボットの最大速度は0.7m/sとした．本手法は不変な箇所がわかればよく，連続的な変化をとらえる必要はないため，本手法ではデータ収集の時間の間隔は一定である必要はない．また，筆者の経験上，安定して自己位置推定に成功している状態において，その推定誤差は0.1m未満と思われることから，0.1m以上の移動すると十分(誤差の範囲ではなく)位置が変化していると考えた．

この方法では，ロボットの位置や傾き，振動による変化と周囲の環境の短期的な変化に対応できる．これについては，ロボットの位置や傾き，振動による変化に関して，図4.10の提案手法による地図において，ロボットの振動により地面の隆起が不安定に計測されていることから確認できる．周囲の環境の短期的な変化に関しては，図4.12のNDTにおける地図に歩行者の計測がもやのように記載されていることに対して，提案手法では歩行者の特徴が消えていることから確認できる．また，間欠的な環境の変化に関しては，自律走行実験の結果から日常的におこる変化程度では特に問題がないと考えられる．

## 4.6 成果

不安定な計測にロバストな絶対位置推定のために，自己位置推定のための地図として，複数回のスキャンデータにおける統計量から求めた尤度場を用いる手法を提案した．前章で述べた手法は，高所の特徴を計測するため，センサの取り付けに制約があることや高所に物体のない環境へ適用できないことが課題としてあげられた．これに対して，計測方法に依存しない方法として，不安定な計測にロバストな自己位置推定の開発に取り組んだ．

提案手法では，自己位置推定のための地図として尤度場を用いる．尤度場を用いることで不安定な計測が生じる環境においても，地図とスキャンデータの一致度を適切に評価する．本研究では，尤度を事前に収集した複数回のスキャ

ンデータの統計量から計算する．この時，参照スキャンとして様々な状況におけるスキャンデータを収集することで，不変な箇所の尤度が高く，不安定な計測が生じた箇所の尤度は低くなる．これにより，一致度の評価において不安定な計測の影響を小さくする．また，尤度の計算に事前に収集した複数回のスキャンデータの統計量を用いることで，静止物の位置などの環境の認識を必要とせずに尤度場を作成できる．このため，不安定な計測のため環境の認識が困難な生活環境においても，適切に尤度を求められる．また，作成した尤度場を用いて，自己位置を推定するためにこの尤度場とスキャンデータの一致度の評価方法について述べた．この一致度の評価をパーティクルフィルタにおけるパーティクルの重みに適用することで，自己位置が推定できる．生活環境の不安定な計測に対する有効性を示すために，他の方法による一致度の評価と比較した．地面の隆起の様な曖昧に計測される対象がある環境と歩行者のいる環境において，占有格子地図との一致度の評価と NDT における評価と比較し，提案手法の有効性を示した．

また，提案手法を用いた自己位置推定により，つくばチャレンジ 2011 のコースにおいて自律ナビゲーションに成功している．つくばチャレンジ 2011 のコースは遊歩道，林間部，広場，市街地歩道，屋内繁華街などの環境がある 1.4km 強のコースであり，多様な環境において自己位置推定が可能であることが確認できる．また，提案手法を用いた自己位置推定による自律ナビゲーションでは，公式な記録をとる本走行においても完走に成功しており，つくばチャレンジ 2011 における本走行完走チームは 66 チーム中 6 チームであり，本手法の有用性を示している．

本手法の課題としては，広場などのセンサの視野内に計測できる物体がない区間では適用できないことがあげられる．この課題に対しては，他のセンサを用いた自己位置推定を並用することが求められる．



## 第5章 オドメトリの系統誤差の軽減

3,4章では、移動ロボットの自己位置推定にとって不可欠となる地図上の絶対位置推定に対して、誤推定の問題を解決する方法を提案した。しかし、これらの手法は、直前の自己位置の推定誤差が十分小さいと仮定して、自己位置を推定する。また、ランドマークが不足する環境には絶対位置推定が適用できないため、オドメトリの誤差が累積し、誤差が増大する。このため、オドメトリの推定精度の向上が求められる。

オドメトリでは左右の車輪のエンコーダの計測をもとにロボットの動作を推定し、これをもとに自己位置を推定する。この時、オドメトリではロボットは実空間を近似した水平な平面上を移動していると仮定して、実空間上の自己位置を推定している。しかし、生活環境での路面は水平な平面ではないため、路面の傾斜や素材により誤差が生じる。ここで、自律ナビゲーションの自己位置推定では地図上の自己位置を推定する。自律ナビゲーションの自己位置推定では地図の歪みは十分小さいとして、オドメトリの推定値を自己位置としている。しかし、歪みの小さい地図の作成は困難なことがあり、長く走行する場合、地図の歪みによる誤差が大きくなることもある。

ここで、オドメトリの誤差は系統誤差と偶然誤差に分けられる。系統誤差は同様に走行すれば、同じ傾向で生じる誤差である。路面の形状や素材や地図の歪みに起因する誤差は、同じ場所を走行すると同様の誤差が生じるため、系統誤差と考えられる。偶然誤差は凹凸のはげしい路面や急加速によるスリップ等が要因である。偶然誤差の予測や計測は困難であるが、一方、系統誤差は同様に走行した際に同じ傾向で生じるため、事前に走行した際に生じた誤差をもとに自律走行時に生じる誤差を推定し、修正できると考えられる。また、系統誤差を軽減する事により、正規分布により適切に誤差の分布を表現可能と考えら

れる．

本章では，自律ナビゲーションにおけるオドメトリの系統誤差を軽減する方法を提案する．事前に走行した際に生じた，地図上のオドメトリの誤差をもとに，オドメトリの系統誤差を推定し，修正する．

## 5.1 オドメトリとその誤差

### 5.1.1 オドメトリの利用

オドメトリは車輪の回転量から自己位置を推定する手法である．車輪に取り付けたエンコーダのカウント値より車輪の回転量を計測する．ロボットは平面上を滑らずに走行していると仮定し，車輪の回転量と車輪径，トレッドのキネマティクスパラメータからロボットの移動量を推定する．この移動量を既知の自己位置から累積することで自己位置を推定する．ここで，推定した移動量には誤差が含まれるため，自己位置の推定結果に誤差が累積する問題がある．この問題に対して地図上の絶対位置推定が用いられる．オドメトリによる自己位置推定がある時点の既知の自己位置からの相対的な位置を推定するのに対して，地図上の絶対位置推定は現時のセンサデータと地図の特徴が一致する絶対位置を推定するため誤差が累積しない．しかし，絶対位置の推定を適用する上でも以下の理由からオドメトリの精度向上は自己位置推定に重要な要素となる．

#### 1. 絶対位置推定の初期位置として利用

多くの絶対位置の推定手法はオドメトリによる推定値を初期位置として局所的に自己位置を探索する．例えば，図 5.1-(a) 左に示すようにロボットの左方向にランドマークが観測されたとする．図 5.1-(a) 右に示すように地図には同様のランドマークがいくつかあるとき，この観測結果のみから自己位置を特定することは困難である．ここでオドメトリの精度が高いと，解はオドメトリによる推定値の近傍にあると考えられる．そこで，オドメトリ推定位置を初期位置として局所的に探索することで自己位置の推定が容易となり，また，誤推定を防ぐことが期待できる．

## 2. 絶対位置推定の誤推定の検出に利用

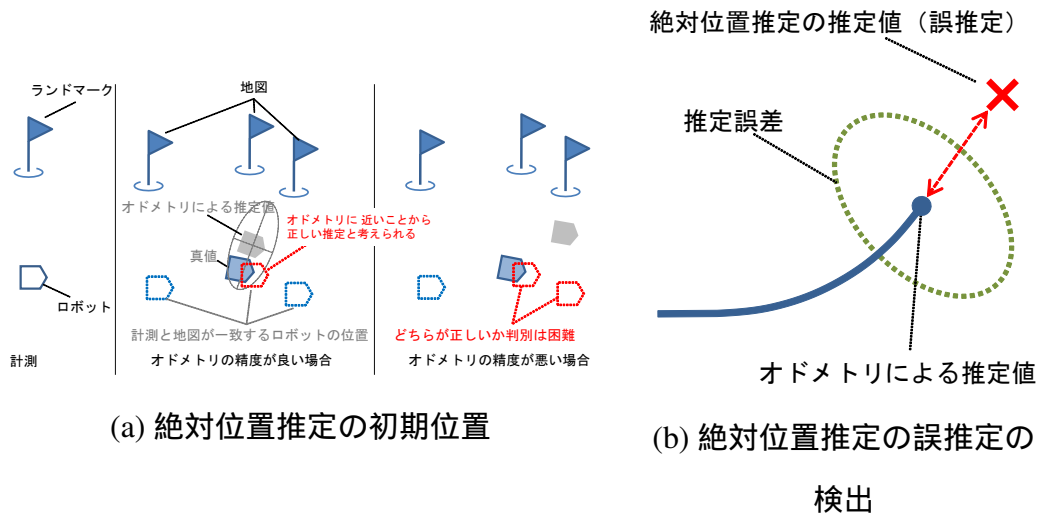
GPS におけるマルチパス誤差や地図とセンサデータの誤った対応付けなど、絶対位置推定の誤りは自己位置推定において致命的な問題となる。このような絶対位置推定の誤りは他の自己位置推定手法の結果と大きく外れることが考えられる。図 5.1-(b) に示すようにオドメトリの推定値を参照として、その誤差分布から絶対位置の推定値が大きく外れた場合、推定の誤りとして検出することができる [13]。しかし、このとき参照となるオドメトリの推定値に大きな誤差があるとその手法は機能しないため、オドメトリの精度が求められる。

3. 絶対位置推定が困難な区間における自己位置推定 実環境においては、人にランドマークの観測を妨げられたり、そもそもランドマークが少ない環境など、センサからランドマークが十分得られないため、絶対位置の推定は環境によっては適用できないことがある。このような区間を長く走行すると、オドメトリにより自己位置の推定誤差が累積する。この時、図 5.1-(c) に示すようにオドメトリの精度が低いと目標経路から大きく外れ、その結果、ランドマークを見失うなどして自己位置推定が破綻することがある。一方、オドメトリの推定精度が高いと、このような区間が長くても走行が可能となる。

このようにオドメトリは移動ロボットの自己位置推定にとって重要な要素となる。

### 5.1.2 オドメトリの誤差

オドメトリでは左右の車輪のエンコーダの計測をもとにロボットの動作を推定し、自己位置を推定している。この時、オドメトリでは、ロボットは実空間を近似した水平な平面上を移動していると仮定して、自己位置を推定する。オドメトリの誤差の原因として、キネマティクスパラメータの誤差がある。オドメトリでは、キネマティクスパラメータとエンコーダの計測値をもとに速度、角



(c) ランドマークが不足する区間の自己位置推定

図 5.1: 自己位置推定におけるオドメトリの役割

速度を求める．キネマティクスパラメータは設計図の値や，実測した値が用いられるが，この値の誤差により，エンコーダより得られる車輪の回転数に対して速度，角速度の推定に誤差が生じる．これはキャリブレーションにより改善することができる．また，路面の傾斜や素材により誤差が生じることがある．オドメトリは滑らかな水平な平面を走行していると仮定しているが，実際には路面には傾斜があったり，素材によっては滑らかではない．このため，路面の傾斜や素材などにより近似による誤差が生じる．この誤差は走行する路面に起因しており，走行する環境に依存した傾向の誤差が生じるため，誤差を軽減することは困難となる．また，凹凸のはげしい路面や急加速などによる車輪のスリップや，小石などの予期しない物への乗り上げにより誤差が生じることもある．

自律ナビゲーションにおける自己位置推定では，地図上の自己位置を推定す

る．オドメトリでは，地図の歪みは十分小さいとして，オドメトリの推定値を地図上の自己位置とする．しかし，実際には，生活環境において歪みの小さい地図を作るとは困難であり，局所的な範囲しか歪みは小さいとみなせないことがある．このため，長く走行すると次第に地図上の自己位置の推定誤差が増大することがある．この地図の歪みは，地図作成時の自己位置の推定誤差が原因であり，地図の歪みによる誤差も路面の傾斜や素材に起因する誤差と同様に，環境に依存した傾向の誤差が生じる．

ここで，オドメトリの誤差には系統誤差と偶然誤差に分けられる．偶然誤差の要因としては，スリップや，小石などへの乗り上げがあげられる．系統誤差の要因としては，キネマティクスパラメータの誤差と路面の素材や形状，地図の歪みがあげられる．偶然誤差は予測や計測が困難である，一方で系統誤差は同様の傾向で生じるため，事前に走行した際に生じた誤差をもとに自律走行時に推定し，修正することが可能と考えられる．これにより，路面の素材や形状と地図の歪みに起因する誤差が軽減し，地図上の自己位置の推定精度の向上することが期待される．また，系統誤差を軽減することで，確率的自己位置推定において重要となる推定誤差の分布を，適切に表現できると考えられる．

## 5.2 オドメトリの修正手法

オドメトリを修正するにあたって，図 5.2 に示すように事前に経路を走行し，この時のセンサデータをもとにオドメトリ誤差地図を作成する．地図上の自己位置を推定し，これとオドメトリを比較することで各区間のオドメトリの誤差(図 5.2 下)を推定する．推定したオドメトリの誤差を地図として，オドメトリの修正する際にはこの地図を参照してオドメトリに生じた誤差を推定し，修正する．

本節では，事前の走行結果をもとにオドメトリ誤差地図を作成方法と，自律走行時にオドメトリ誤差地図を用いてオドメトリを修正する方法について述べる．

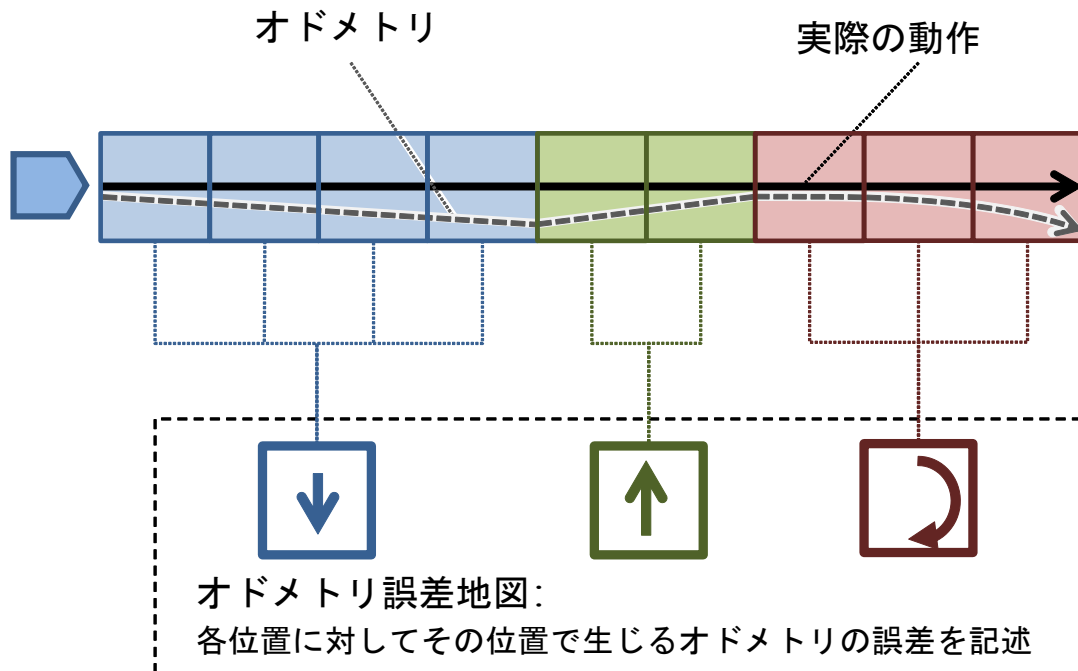


図 5.2: オドメトリ誤差地図

### 5.2.1 オドメトリ誤差地図の作成

オドメトリ誤差地図は、事前に人がロボットを操作するなどして経路を走行した際のセンサデータから作成する。まず、地図の状態空間を定義する。地図の状態空間は系統誤差に影響を与える要素により定義する。これらの要素は走行経路や走行の方法によって決める。例えば、オドメトリの系統誤差の要因として路面が考えられる。一本道の経路を走行する場合、走行している路面すなわち位置は走行距離によって示すことができるため、地図の空間は走行距離を要素とした1次元とする。実空間を近似した2次元平面上で走行の指示をする場合、地図の空間は2次元平面上の位置とロボットの進行方向の3次元空間とする。また、速度の変更が激しい走行が求められ、さらに速度によって系統誤差が変化する環境の場合、速度も要素に追加することも考えられる。

オドメトリ誤差地図は定義した状態空間を格子状に区切る。各セルには単位距離走行あたりのオドメトリの推定誤差の平均値を記録する。これは地図上のオド

メトリの推定誤差であり，事前にロボットが走行した際のデータより求める．セル  $a$  における単位距離走行あたりのオドメトリの誤差の平均値  $d\mathbf{e}_a = (\overline{dx_a}, \overline{dy_a}, \overline{d\theta_a})^T$  はセル  $a$  内で生じたオドメトリの誤差の総和  $\mathbf{e}_a$  と走行距離  $r_a$  より求める．

$$d\mathbf{e}_a = \frac{\mathbf{e}_a}{r_a} \quad (5.1)$$

走行距離はオドメトリにより得られる．地図上のオドメトリの推定誤差について，地図上の自己位置とオドメトリによる推定値の差である．これは，地図作成時には，地図作成時の自己位置の推定値とオドメトリによる推定値の差であり，既知の地図に対してオドメトリ誤差地図を作成する場合は，地図上の自己位置の推定結果とオドメトリによる推定の差を用いる．地図上の自己位置の推定値  $\hat{\mathbf{p}}$  に対するオドメトリによる自己位置の推定値  $\hat{\mathbf{q}}$  の差を誤差とする．既知の地図に対して，地図上の自己位置を推定する方法は GPS やレーザスキャナを用いたスキャンマッチングなど様々な手法があるが，オドメトリより信頼できる推定値である必要がある．時刻  $t$  におけるオドメトリの誤差  $\mathbf{e}_t$  は以下となる．

$$\mathbf{e}_t = \hat{\mathbf{q}}_t - \hat{\mathbf{p}}_t \quad (5.2)$$

また，地図上のオドメトリの推定誤差はある程度走行するたびに計測し，同時に自己位置を修正する．これは，姿勢の誤差に起因する位置の誤差を小さくするためであり，姿勢の誤差が大きくなる前に自己位置を修正する必要がある．オドメトリによる推定値  $\hat{\mathbf{q}}_{t+1}$  はオドメトリより信頼できる自己位置推定の結果  $\hat{\mathbf{p}}_t$  から求める．

$$\hat{\mathbf{q}}_{t+1} = \hat{\mathbf{p}}_t + \delta_t \quad (5.3)$$

ここで， $\delta_t$  は時刻  $t$  から  $t+1$  までのオドメトリによる移動量の推定値である．

セル  $a$  内で生じたオドメトリの誤差の総和  $\mathbf{e}_a$  はロボットの状態  $s$  がセル  $a$  内である時の誤差の総和とする．

$$\mathbf{e}_a = \sum_{s_t \in a} \mathbf{e}_t \quad (5.4)$$

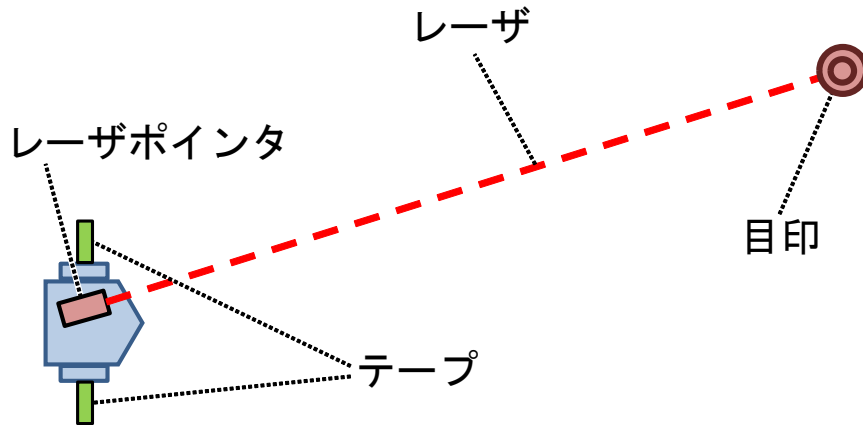


図 5.3: ロボットを初期位置に設置する方法

### 5.2.2 オドメトリの修正

作成したオドメトリ誤差地図をもとに，自律走行時にオドメトリの誤差を修正する方法について述べる．オドメトリ誤差地図に記された単位走行距離あたりのオドメトリの誤差と，走行距離よりオドメトリの誤差を推定する．走行距離  $r$  はエンコーダのカウント値をもとにオドメトリにより得る．ロボットの状態をもとにオドメトリ誤差地図を参照し，その状態における単位走行距離あたりのオドメトリの誤差の平均値  $de$  を得る．これらよりオドメトリの誤差  $e$  を次式の通り推定する．

$$e = r \times de \quad (5.5)$$

修正した自己位置  $\hat{x}_c$  は推定した誤差  $e$  とオドメトリによる推定値  $\hat{x}$  より次式の通り計算する．

$$\hat{x}_c = \hat{x} - e \quad (5.6)$$

以上の処理をオドメトリの更新の度に実行し，自己位置を修正する．



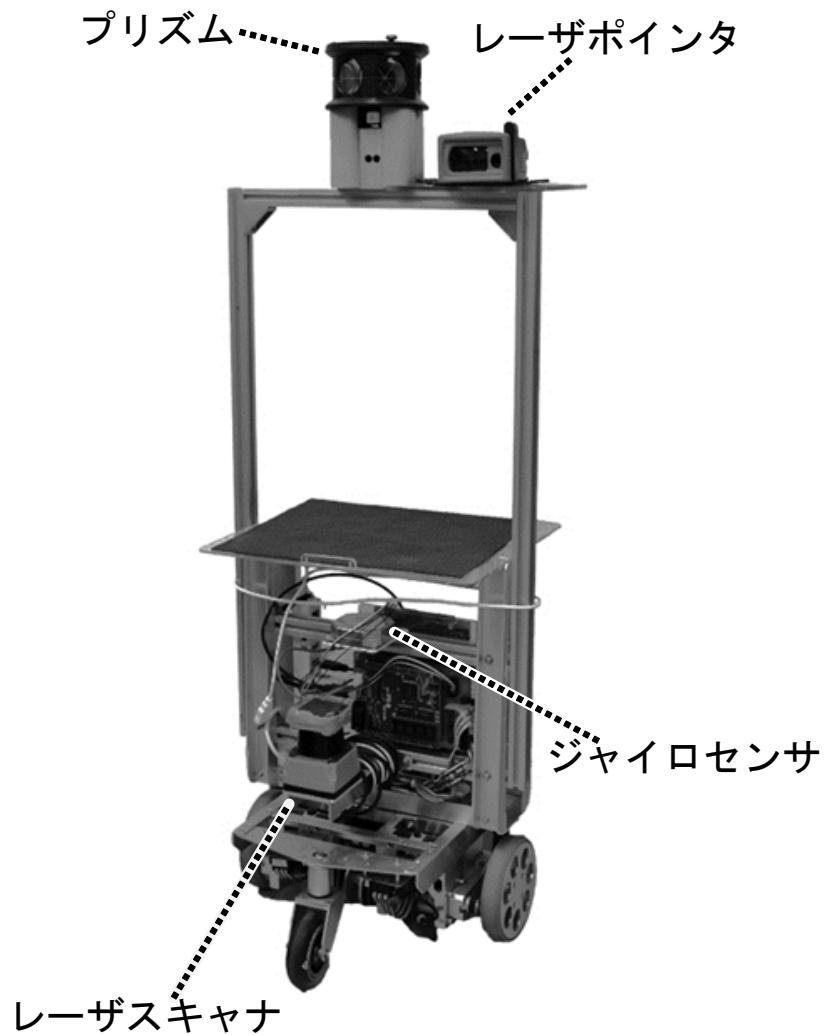


図 5.4: ロボットの外観

### 5.3 評価実験

オドメトリ誤差地図を用いて修正したオドメトリにより，系統誤差が軽減できるかを評価するために実験を行った．実験では系統誤差の生じるいくつかの環境において，オドメトリによる自律走行と誤差地図により修正したオドメトリによる自律走行の走行軌跡を比較した．ロボットに地図上で目的の位置までの走行を指示し，地図上の指定した位置と対応する実空間上の位置に到達することを確認する．この時，歪んだ地図上と実空間上で対応する位置を求めることは困難であり，実験を容易とするため，歪みの小さい地図を作成することに



図 5.5: 実験環境:横断勾配



図 5.6: 実験環境:タイルカーペット

した．地図の作成にはレーザスキャナを用い，ロボットを移動させずに計測したスキャンデータから地図を作成した．この地図は，ロボットの移動しないため，地図作成時の自己位置の誤差はなく，レーザスキャナの視野の範囲内では歪みが小さい．レーザスキャナには北陽電機社製 UTM-30LX を用いた．UTM-30LX は視野が距離 30m，角度 270 度である．この方法により，地図の空間は実空間と十分近いと考えられる，実験では実空間上でロボットが指示した通りの位置に到達することを確認した．

自律走行時ロボットには一定の距離直進し，静止する指示を行った．この時，ロボットは自己位置の推定値をもとに初期位置から目標位置を結ぶ直線上を走行し，指示した位置で静止するように制御している．このため，ロボットは自己位置の推定値がおよそ初期位置 ( $x = 0.00, y = 0.00$ ) から目標位置を結ぶ直線上となるように走行し，推定値が目標位置となったときに制止する．自律走行時の軌跡がまっすぐであり，指示した距離で静止していれば，自己位置推定の誤差が小さいといえる．ロボットの走行軌跡の測定には自動追尾トータルステーション(トプコン社製 GPT-9000A)を用いた．自動追尾トータルステーションによりロボットに搭載したプリズムを追尾，測定し，これをロボットの走行軌跡とした．ここで，本実験では誤差地図作成のためとオドメトリによる自律走行，オドメトリを修正しながら自律走行の3回同じ経路を走行させるため，各走行の初期位置を一致させる必要がある．そこで，図 5.3 に示すように，初期位置におけるロボットの左右の車輪の位置を路面に貼ったテープであわせ，ロボット上に固定したレーザポインタが指す位置を遠方に設置したマーカーに一致さ

せることで向きをあわせた．走行経路は直進のみであるため，地図の空間は走行距離を要素とした1次元とした．セルのサイズは1mとし，地図の空間を等間隔に区切った．レーザスキャナを用いたスキャンマッチングによる自己位置推定結果とオドメトリの差を誤差とした．スキャンマッチングにはNDTの評価関数を用い，これをもとにモンテカルロ法により最適解を探索する．1.25cm走行するごとに誤差を計測し，この時得られたスキャンマッチングによる自己位置推定結果により自己位置を修正する．ロボットが10m走行した際の自己位置をこの手法により推定したところ，トータルステーションの計測値と比較した結果，最大で7cm程度のずれがあった．

実験には独立二輪駆動型の移動ロボット(図5.4,縦36cm,横35cm,高さ81cm)を用いた．誤差の計測のスキャンマッチングにはレーザスキャナとして北陽電機社製 UTM-30LX を用いた．また，ロボットの上部にはトータルステーションによる位置計測のためのプリズムを搭載している．実験は系統誤差の生じる環境として舗装路面の横断勾配の往路、復路とタイルカーペット(それぞれ写真を図5.5, 5.6に示す)を対象に実験を行った．なお，いずれの実験においても，ロボットは原点で $x$ 軸方向を向いた状態を初期位置とし， $y$ 軸は左方向，上から見て反時計回りを姿勢の正方向とした．

### 5.3.1 舗装路面の横断勾配

#### 往路

実験では目標の静止位置を  $x = 10\text{m}, y = 0.0\text{m}$  とし，図5.5の矢印の箇所を10m直進し，静止を指示した．ここは，道路を横断する方向へおよそ2度の勾配が設けられている．図5.7に作成された誤差地図のデータを示す．各グラフは横軸に走行距離，縦軸に各成分の1m走行あたりのオドメトリの誤差を示し，(a)は位置の $x$ 軸成分，(b)は位置の $y$ 軸成分，(c)は姿勢の誤差である．誤差地図には経路全体において姿勢の推定に正方向の大きな誤差が生じることが記録され，この経路上ではオドメトリによる自己位置の推定が左方向にそれていくことを示している．

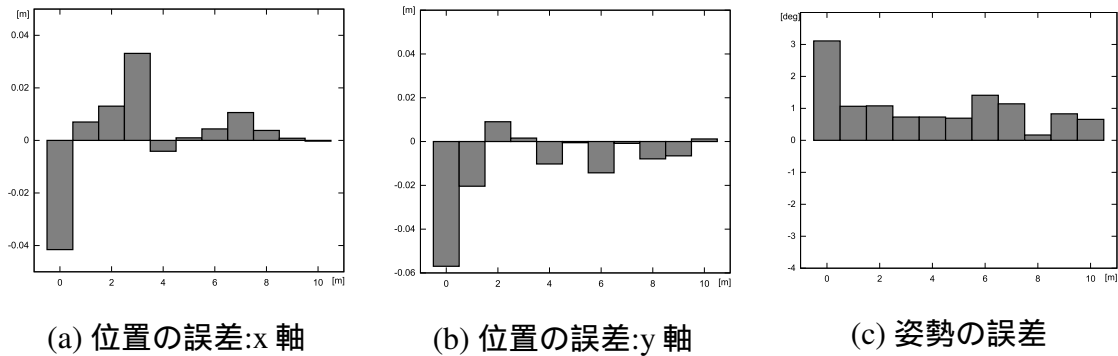


図 5.7: 横断勾配往路におけるオドメトリ誤差地図

オドメトリによる走行とオドメトリを修正しながら走行を各 15 回行った結果として図 5.8 にロボットの走行軌跡，図 5.9 に静止位置を示す．図 5.8 では通常のオドメトリによる自律走行時の軌跡を破線で，本手法によりオドメトリを修正しながら走行した時の軌跡を実線で示す．図 5.9 では，通常のオドメトリによる自律走行時の静止位置を丸で，本手法によりオドメトリを修正しながら走行した時の静止位置を十字で示す．オドメトリによる姿勢の推定の誤差により推定値が左方向へそれていくため，オドメトリによる自律走行では，ロボットの走行軌跡は指示した経路 (10m の直線) から右方向に大きくそれる傾向がある．一方，本手法よりオドメトリを修正しながら走行した場合，およそ直進することに成功し，静止位置も目標である  $x = 10\text{m}, y = 0.0\text{m}$  の位置付近に分布している．このことから，本手法により姿勢の系統誤差を修正できたことがわかる．各走行の静止位置のばらつきは偶然誤差および初期位置のずれが原因と考えられる．また，静止位置の平均値もオドメトリの  $x = 9.92\text{m}, y = -8.6\text{m}$  と比較して， $x = 10.01\text{m}, y = -0.05\text{m}$  と目標に十分近いことから，表 5.2 に自己位置の推定誤差の平均値を示す．オドメトリの推定誤差は x 軸方向に  $0.079\text{m}$ ，y 軸方向に  $-0.860\text{m}$  であり，姿勢に大きな誤差が生じるため走行軌跡がそれて y 軸方向に大きな誤差が生じている．提案手法では x 軸方向に  $-0.018\text{m}$ ，y 軸方向に  $-0.056\text{m}$  と，誤差が大きく軽減していることが確認できる．

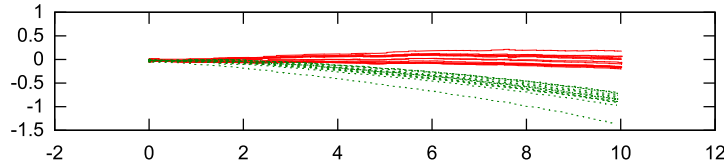


図 5.8: 横断勾配往路における走行軌跡

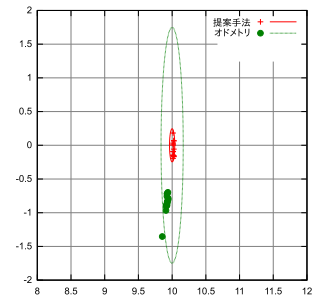


図 5.9: 横断勾配往路における静止位置

表 5.1: 横断勾配往路における推定誤差の平均値

	x	y
目標位置	10.00m	0.00m
オドメトリの誤差	0.079m	-0.860m
提案手法の誤差	-0.018m	-0.056m

### 復路

往路と反対の向きの経路において実験を行った．往路と同様に目標の静止位置を  $x = 10\text{m}, y = 0.0\text{m}$  とし 10m 直進し，静止を指示した．図 5.10 に作成された誤差地図のデータを示す．誤差地図には姿勢の推定に負方向の大きな誤差が生じることが記録されている．このことから，同じ経路においても走行する向きにより誤差が異なることが確認できる．

オドメトリによる走行とオドメトリを修正しながら走行を各 15 回行った結果として図 5.11 にロボットの走行軌跡，図 5.12 に静止位置を示す．オドメトリによる自律走行では，指示した経路に対して左にそれる傾向がある．いずれの場合もロボットの走行軌跡は指示した経路 (10m の直線) から右方向にそれたが，本手法よりオドメトリを修正しながら走行した場合，往路と同様に姿勢の系統誤差を修正し，およそ真っ直ぐ走行することに成功している．しかし，静止位置は  $x = 10.05\text{m}$  付近に分布しており  $x$  軸方向に対して系統的な誤差が生じている．これは，誤差地図の作成時のスキャンマッチングによる位置推定の誤差に

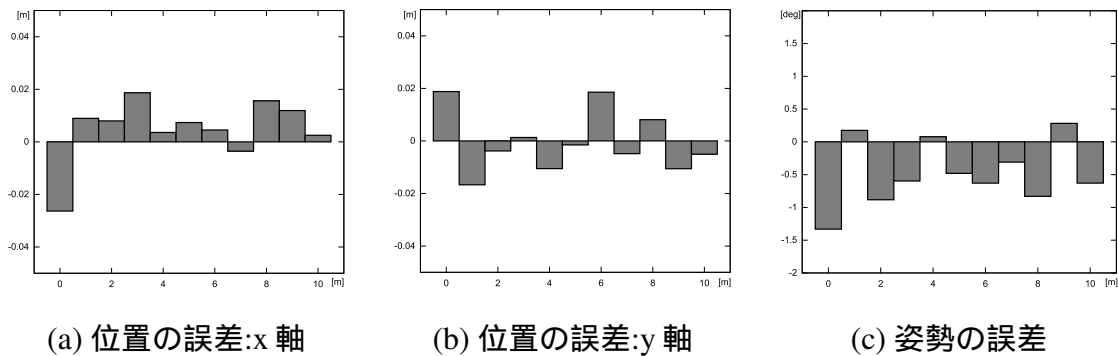


図 5.10: 横断勾配復路におけるオドメトリ誤差地図

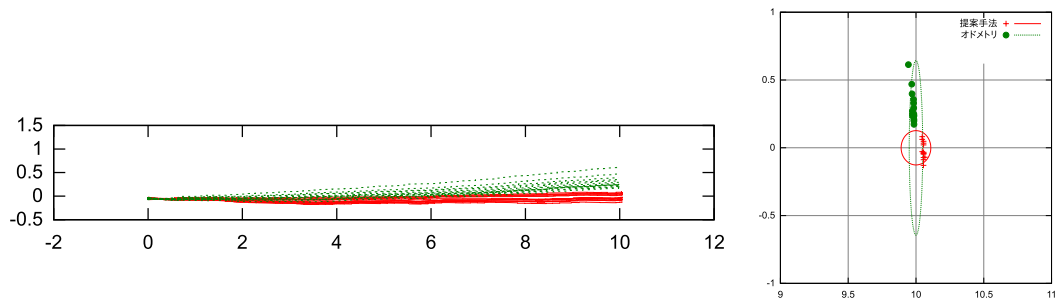


図 5.11: 横断勾配復路における走行軌跡

図 5.12: 横断勾配復路  
における静止位置

より，オドメトリ誤差地図に誤りが生じたこと原因と考えられる．表 5.1 に自己位置の推定誤差の平均値を示す．オドメトリの推定誤差は x 軸方向に  $-0.022\text{m}$ ，y 軸方向に  $-0.301\text{m}$  であり，姿勢に誤差が生じた結果，y 軸方向に大きな誤差が確認できる．提案手法では x 軸方向に  $0.054\text{m}$  x 軸方向に，y 軸方向に  $0.021\text{m}$  であった．姿勢の系統誤差が軽減できたため，y 軸方向の誤差が軽減している．しかし，x 軸方向の誤差が軽減できていない．これはオドメトリ誤差地図の作成時の地図上の位置の推定に誤差が生じたためと考えられる．また，往路とは誤差の傾向が異なることから，これらの誤差はキャリブレーションや地図をデッドレコニングに合わせて歪める方法では対応できないことがわかる．

### 5.3.2 タイルカーペット

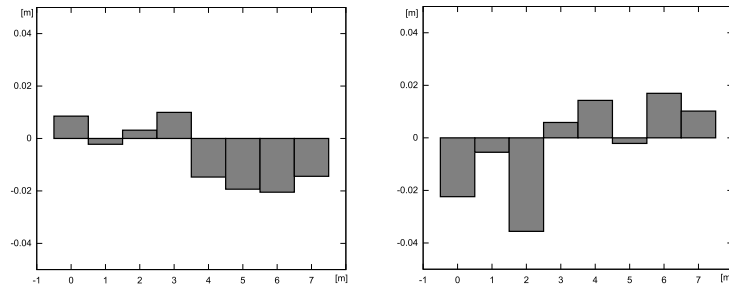
タイルカーペット上では，ロボットの左右の車輪が異なるタイル上にある時，左右の車輪はそれぞれのタイルの目に応じた影響を受ける．このため，オドメ

表 5.2: 横断勾配復路における推定誤差の平均値

	x	y
目標位置	10.00m	0.00m
オドメトリの誤差	-0.022m	-0.301m
提案手法の誤差	0.054m	0.021m

トリによる姿勢の推定に系統的ではない誤差が生じる．そこで，この実験ではジャイロセンサを用いて，ジャイロセンサから得られる角速度より姿勢を推定することとした．また，自律走行時，誤差地図による姿勢の修正は行わず，位置のみ修正し，これを評価した．

タイルカーペット上の実験では，目標の静止位置を  $x = 7\text{m}, y = 0.0\text{m}$  とし，7m 直進する経路を走行した．図 5.13 に作成した誤差地図のデータを示す．誤差地図は経路のほぼ全体において，一般的には小さいと考えられる進行方向 ( $x$  軸) への誤差が，経路を 4m 走行したのちに 1m の走行に対し 2cm 前後の誤差が生じていることを示している．オドメトリによる走行とオドメトリを修正しながら走行を各 20 回行った結果として図 5.14 にロボットの走行軌跡，図 5.15 に静止位置，この環境では並進方向に系統誤差が生じ，オドメトリによる自律走行の静止位置が， $x = 7.02\text{m}$  付近に分布している．一方，本手法よりオドメトリを修正しながら走行した場合  $x = 7.00\text{m}$  付近に分布していることから，この位置の系統誤差を軽減していることがわかる．表 5.3 に自己位置の推定誤差の平均値を示す．この時，自己位置の推定誤差はオドメトリでは  $x$  軸方向に  $-0.026\text{m}$  の誤差が生じている．オドメトリの進行方向への誤差は一般的に小さいため，長く走行すると修正が困難になることがある．提案手法では， $x$  軸方向への推定誤差はわずか  $0.002\text{m}$  であり，位置に対する誤差が軽減できていることが確認できる．



(a) 位置の誤差:x 軸

(a) 位置の誤差:y 軸

図 5.13: タイルカーペットにおけるオドメトリ誤差地図

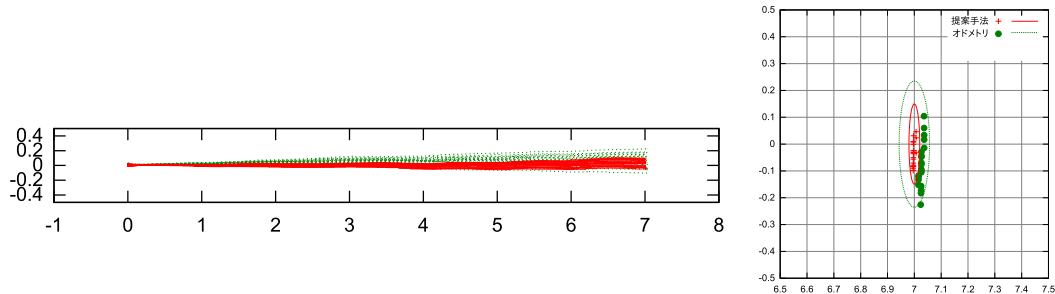


図 5.14: タイルカーペットにおける走行軌跡

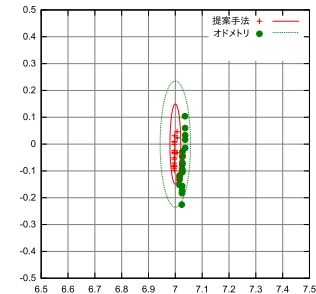


図 5.15: タイルカーペットにおける静止位置

表 5.3: 横断勾配復路における推定誤差の平均値

	x	y
目標位置	10.00m	0.00m
オドメトリの誤差	-0.026m	-0.07m
提案手法の誤差	0.002m	-0.026m

## 5.4 検討

ここでは、本手法を実用するための問題に対する検討について述べる。本手法は事前に走行した際のデータから得られる統計量をもとに誤差を推定し、自己位置を修正する。このため、標本となるオドメトリの誤差のデータ収集は本手法において重要な要素となる。そこで、データ収集に密接に関連する誤差計測の間隔、セルのサイズについて検討する。



また，本手法は誤差の計測にオドメトリ以外の自己位置推定手法を必要とするため，オドメトリ以外の自己位置推定が困難な環境への適用は難しい．しかし，この問題は自己位置推定が可能な環境を一時的に用意すれば解決できる．そこで，オドメトリ以外の自己位置推定の困難な環境において本手法を適用した例とその結果についても述べる．

#### 5.4.1 オドメトリの誤差計測の間隔

誤差の計測間隔は以下の2点の理由から短い間隔であることが望ましい．1つは姿勢の誤差に起因する位置の誤差を小さくするためである．本手法では誤差地図に記録されたオドメトリの誤差は各成分が独立と仮定している．そこで，姿勢の誤差に起因する位置の誤差を十分小さくするため，誤差の計測を行うとともに，この時求めたオドメトリより信頼できる推定値で自己位置を修正する．この自己位置の修正は姿勢の誤差が大きくなならないよう，短い間隔で繰り返す必要がある．

もう1つは，誤差地図の離散化の影響を小さくするためである．本手法では誤差地図を格子状に区切っているため，セルをまたがって走行した瞬間のデータは適切に反映されない．このため，セルのサイズに対して誤差の計測間隔を小さくし，セルをまたがって走行した際のデータの影響を小さくすることが望ましい．

実験では1.25cm 走行ごとに誤差を計測した．1.25cm 走行する間に生じる誤差は極めて小さく，姿勢の誤差に起因する位置の誤差は無視できる．また，セルのサイズ1mと比較しても十分小さく，離散化の影響も小さいといえる．

#### 5.4.2 セルのサイズ

セルのサイズは前述のように誤差地図の離散化の影響を小さくするため，誤差の計測間隔に対して十分大きいことが望ましい．一方，同セルを走行している間に路面が変化するなどによって系統誤差が変化すると，誤差地図によるオ

ドメトリの誤差の推定に誤りが生じ、適切にオドメトリを修正できない。これは、誤差地図に記録されたオドメトリの誤差は同セル内で一様としていることが原因であり、セルのサイズは系統誤差の変化に対して小さいことが好ましい。そこで、実験ではセルのサイズを1mとした。これは1m 走行する間に生じるオドメトリの誤差は小さく、この区間を一様な路面と近似しても問題ないと考えたためである。

### 5.4.3 オドメトリ以外の自己位置推定が困難な環境

本手法はオドメトリ以外の自己位置推定手法が困難な環境への適用は原理的に難しいが、この問題は一時的にオドメトリ以外の自己位置推定が可能な環境を用意することができれば解決できる。ここでは5.3 章と同様にスキャンマッチングによる誤差地図作成を、スキャンマッチングが困難な環境において適用した例について述べる。図 5.16 に実験環境の写真を示す。この環境では、周囲にランドマークとなる静止物は少なく、また、路面に傾斜のある区間があるため水平に取り付けたレーザスキャナにより、時折路面が観測され、スキャンマッチングが困難である。そこで、誤差地図作成のためのデータを収集する時のみ、一時的にランドマークとなる板を設置して誤差地図を作成した。

この環境でロボットには 20m 直進を指示した。図 5.17 に作成した誤差地図のデータを示す。本環境では舗装路面の横断勾配のためオドメトリによる自己位置の推定が左方向にそれていく。図 5.18 に自律走行時の走行軌跡、Tab.5.4 に自律走行時のロボットの静止位置を示す。オドメトリによる自律走行時、オドメトリの誤差のためロボットの走行軌跡は指示した経路から大きくそれた。ロボットが路肩に乗り上げ、走行が危険であったため、 $x = 13.44\text{m}$ 、 $y = -1.66\text{m}$  の位置で停止させた。誤差地図によりオドメトリを修正しながら走行した際の軌跡は明らかな改善が見られ、 $x = 20.07\text{m}$ 、 $y = 0.63\text{m}$  の位置で静止している。

この結果により本手法は誤差地図の作成時に一時的に自己位置推定が可能な環境を用意することで、実際にはオドメトリ以外の自己位置推定が困難な環境でも適用可能であることがわかる。



図 5.16: スキャンマッチングが困難な環境

表 5.4: スキャンマッチングが困難な環境における静止位置

	x	y
目標位置	20.00m	0.00m
オドメトリ	13.44m	-1.66m
提案手法	20.07m	0.63m

## 5.5 成果

オドメトリ誤差地図に基づき自己位置を修正する手法と、その評価実験、実用のための検討について述べた。提案手法は事前に人がロボットを操作して走行予定の経路付近を走行させ、この時生じたオドメトリの誤差を計測する。この誤差の計測結果をもとに誤差地図を作成し、自律走行時には誤差地図を参照することでオドメトリの誤差を推定し、自己位置を修正する。

評価実験では、本手法により系統誤差を軽減し、オドメトリの精度が向上す

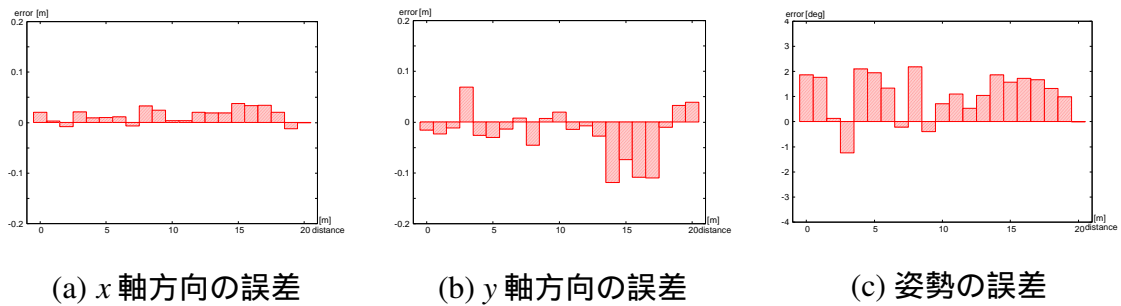


図 5.17: スキャンマッチングが困難な環境におけるオドメトリ誤差地図

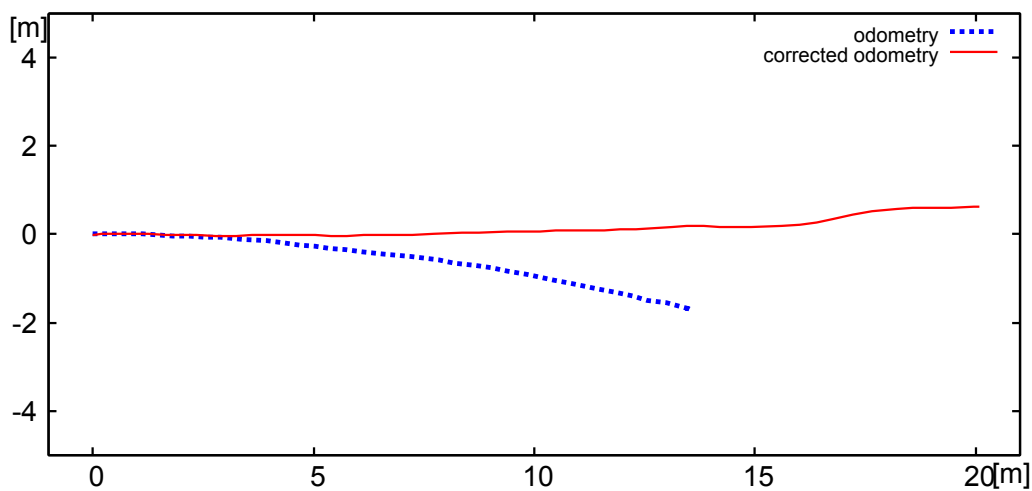


図 5.18: スキャンマッチングが困難な環境における走行軌跡

ることを評価した．複数の異なる環境においてオドメトリの自己位置に基づく自律走行の軌跡と，提案手法によりオドメトリを修正しながら自律走行した際の軌跡を比較した．結果，提案手法により系統誤差を軽減し，オドメトリの精度が向上することを確認した．

提案手法はオドメトリを用いているあらゆる移動ロボットに対して有効で，それらの運用場面で環境に応じたデータを予め所得しておくことで精度良いオドメトリを得ることを可能とする．また，地図上の自己位置をもとに走行する多くのロボットは予め経路を走行させて環境データ取得する必要があるため，本手法は利用するにあたって特別な操作を必要しない．以上のように，本手法は手軽に利用でき，多くのロボットにおいてオドメトリの精度を向上することが期待できる．

本手法の課題としては，オドメトリ誤差地図作成のため，データ収集があげ

られる．オドメトリ誤差地図の作成のため，事前に経路を走行した際のオドメトリには偶然誤差が含まれる．このため，十分な量のデータが必要となるが，どの程度の量のデータが必要かについては十分に検討できていない．

## 第6章 結論

本研究では，生活環境における自律ナビゲーションのために，様々な環境に適用可能でロバストな自己位置推定の実現に取り組んだ．生活環境における自己位置推定について考察し，絶対位置推定の誤推定とランドマークの不足による絶対位置推定が不可能な状況が問題であると分析した．考察の結果から，絶対位置推定とデッドレコニングのそれぞれの改善に取り組むこととした．

絶対位置推定ではランドマークの不足の問題に対して，様々な環境で容易に観測できるランドマークによる自己位置推定が求められる．そこで，外界センサにレーザスキャナを用いて計測全体をランドマークとする枠組みを採用した．また，絶対位置推定の誤推定の原因は，歩行者の移動などの時間による変化や，日照やロボットの振動などの計測の条件の変化のために生じる不安定な計測である．そこで，外界センサにレーザスキャナを用いて計測全体をランドマークとする枠組みをもとに，不安定な計測に有効な2つの手法を提案した．

1つは計測を工夫する手法で，高所の特徴を用いるスキャンマッチングである．高所特徴を用いたスキャンマッチングでは，上方に傾けたレーザスキャナにより人の頭上より高所の特徴を計測する．生活環境における環境の変化は主に人の活動によるものであり，不安定な計測が少ないと考えられる人の背より高い位置の特徴を用いることとした．また，人の頭越しに計測することで，人にランドマークの観測を妨げられず，ランドマークの不足の問題に対しても有効である．提案手法では，オドメトリをもとに複数のスキャンデータを重ね合わせ，特定の高さの範囲の特徴を用いることで自己位置推定に有効な特徴を得る．この特徴を用いてICPスキャンマッチングにより，自己位置を推定する．実験ではつくばチャレンジ2009,2010にて実際の公園，遊歩道，林間部，市街地などを含む1km強の経路の自律ナビゲーションに成功し，本手法の有効性を示し

た．また，高所の形状特徴が人の生活環境における自己位置推定にとって，有効な特徴であることを示した．しかし，高所を計測するため，高所に物体のない区間には適用できないことや，センサの取り付けに制約があることが課題としてあげられた．

もう1つは，統計量に基づくレーザスキャナの尤度場を用いた自己位置推定を提案した．これは，高所の特徴を用いるスキャンマッチングの手法の課題に対して，計測方法に依存しない方法として提案した．不安定な計測による誤推定は，不安定な計測が偽のランドマークとなることで起こる．そこで，地図として反射点観測の尤度を用いることで，不安定な計測の影響を小さくする．反射点観測の尤度とは，ある位置で反射点が観測されたときに，その観測がロボットが走行している環境で得られることが尤もらしいかを示す値である．複数回計測を重ねた時，不安定な計測はバラつくと考えられるため，尤度が低くなる．本研究では，複数回のスキャンデータの統計量をもとに尤度場を作成する方法を提案した．複数回のスキャンデータの統計量をもとに尤度場を計算することで，静止物の位置や不安定な計測の原因の認識を必要とせず，尤度を計算できる．この尤度場を地図としてスキャンデータとの一致度の評価する．この評価値と他の手法を比較し，不安定な計測が生じる環境でもロボットの位置，姿勢の評価が適切にできることを示した．また，この評価値を，パーティクルフィルタのリサンプリングにおけるパーティクルの重み付けに用いることで，ロバストな自己位置推定を実現した．実証実験ではつくばチャレンジ2011にて実際の公園，遊歩道，林間部，市街地，屋内繁華街などを含む1.4kmの経路の自律ナビゲーションに成功し，本手法の有効性を示した．

デッドレコニングの精度向上では，車輪型移動ロボットのデッドレコニングであるオドメトリを対象に，系統誤差を軽減する方法を提案した．系統誤差は同様に走行した際に，同様の傾向で生じることが期待できる．ことから，事前に走行した際のセンサデータより，自律走行時に生じる誤差を推定し，修正する．また，系統誤差の生じる環境において，オドメトリと比較して自己位置の推定精度が向上することを実験により確認した．以上により，様々な環境で有効に働くロバストな自己位置推定を実現し，実際の様々な生活環境においてロ

ロバストな自律ナビゲーションのための自己位置推定を実現した．

人の生活環境で移動ロボットの自律ナビゲーションを実現するには，ロバストな自己位置推定が重要である．本研究は，人の生活環境における自己位置推定の課題を考察し，解決する方法を提案した．また，実際に提案手法を用いて，つくば駅付近の市街地歩道 1km 強の自律ナビゲーションを実現している．このことから，本研究は人と共生する移動ロボットの研究開発に対して有用である．また，本研究の提案手法では、少ないセンサでロバストな自己位置推定が実現できることから、多くの車輪型移動ロボットに適用できロボット産業の発展に役立つことが期待できる．



## 謝辞

本研究は，博士(工学) 大矢晃久 筑波大学システム情報系教授のご指導のもと行われたものです．同教授には本研究を進めるにあたり，ご多忙にあるにも関わらず，研究上の議論，論文の執筆，公道での実証実験等にお付き合いいただきました．ここに感謝の意を表します．

また，本論文をまとめるにあたり，適切なご助言をくださった博士(工学) 安永守利 筑波大学システム情報系教授、博士(工学) 滝沢穂高 筑波大学システム情報系准教授に感謝致します．

工学博士 油田信一 芝浦工業大学工学部電気工学科特任教授，ならびに工学博士 坪内孝司 筑波大学大学院システム情報工学研究科教授には研究を進める上で数多くの有益なご指導とご援助を承りました．ここに感謝の意を表します．

筑波大学知能ロボット研究室のメンバーの皆様には、研究生活の様々な面でお世話になりました．坂東茂氏，渡辺敦志氏，原祥堯氏には本研究を進める上で，様々なご助言をいただくとともに，多くのことを学ばせていただきました．識名拓氏(現株式会社安川電機)，木下和樹氏(現任天堂株式会社)ならびに，吉田森彦氏，松本祥氏，高木勇武氏にはつくばチャレンジのチームメンバーとして自律ナビゲーションロボットの作成に多大なご協力をいただきました．また，田所裕貴氏(現トヨタ自動車株式会社)には研究活動における様々な場面でご助力いただきました．お世話になったすべての方のお名前をあげることはできませんが，多くの方々にご協力をいただいたこと心より感謝致します．

最後に，ありとあらゆる場面で私を温かく見守り続けてくれた両親には深く感謝しております．

上述の皆様をはじめ，協力していただいた皆様へ心から感謝の気持ちと御礼を申し上げます．

## 参考文献

- [1] ”プロトタイプロボット展のために開発されたすべてのロボットの紹介”, 日本ロボット学会誌, Vol.24, No.2, pp.25-58, 2006.
- [2] ”自動運転車,公道で初実験 首相「世界一の技術」: 日本経済新聞社”, 日本経済新聞社.(オンライン), [http://www.nikkei.com/article/DGXNASFS0900\\_Z01C13A1PE8000/](http://www.nikkei.com/article/DGXNASFS0900_Z01C13A1PE8000/), (参照 2013-11-12)
- [3] 油田 信一: ”日本チャレンジャー趣旨と課題”, 第8回計測自動制御学会システムインテグレーション部門講演会 (SI2007), 3C1-1, 2007.
- [4] ”つくばチャレンジ”, <http://www.tsukubachallenge.jp>, (参照 2013-12-02).
- [5] 青山 元: ”屋外清掃ロボット スバルロボハイター RS1 の開発”, 日本ロボット学会誌, Vol.24, No.2, pp.153-155, 2006.
- [6] 神田 真司, 村瀬 有一, 沢崎 直之, 麻田 務: ”サービスロボット「enon」の開発”, 日本ロボット学会誌, Vol.24, No.3, pp.288-291, 2006.
- [7] 日浦 亮太: ”wakamaru の挑戦”, 日本ロボット学会誌, Vol.24, No.3, pp.284-287, 2006.
- [8] 細田 祐司: ”人間共生ロボットの開発への取り組み”, 日本ロボット学会誌, Vol.24, No.3, pp.296-299, 2006.
- [9] 下笹 洋一, 若林 潔, 有木 孝夫, 大町 利夫, 末富 大剛, 杉浦 正則, 小谷 健太郎: ”警備ロボット“ ガードロボ ”による警備サービス.” 日本ロボット学会誌, Vol.24, No.3, pp.308-311, 2006.

- [10] 柴田 崇徳: ”メンタルコミットロボット・パロとロボット・セラピーの展開”, 日本ロボット学会誌, Vol.24, No.3, pp.319-322, 2006.
- [11] 久野 義徳: ”サービスロボットのための視覚と対話の相互利用”, 情報処理学会論文誌 コンピュータビジョンとイメージメディア, Vol.47, pp.22-34, 2006.
- [12] 竹内 栄二郎, 山崎 将史, 田中 一志, 大野 和則, 田所 諭: ”複数の外界センサを用いた位置推定モジュール群による屋外環境における自己位置推定”, 日本ロボット学会誌, Vol.30, No.3, pp.296-304, 2012.
- [13] 伊達 央, 滝田 好宏: ”自律移動ロボット Smart Dump による実世界環境走行実験”, 日本ロボット学会誌, Vol.30, No.3, pp.305-313, 2012.
- [14] 江口 純司, 尾崎 功一: ”DGPS 測位点の精度評価に基づく占有格子地図の作成手法の構築”, 日本機械学会論文集 C 編, Vol.794, No.78, pp.3459-3468, 2012.
- [15] 富沢 哲雄, 村松 聡, 平井 雅尊, 佐藤 晶則, 工藤 俊亮, 末廣 尚士: ”グリッドマップのマッチングに基づく未知障害物にロバストな自己位置推定”, 日本ロボット学会誌, Vol.30, No.3, pp.280-286, 2012.
- [16] 石田 卓也, 関口 誠, 岡村 公望, 福永 和海, 大矢 晃久: ”富士ソフト/筑波大学 MRIM プロジェクトのつくばチャレンジ 2009 参加レポート”, つくばチャレンジ 2009 開催記念シンポジウム参加レポート集, pp.26-27. 2009.
- [17] 友納 正裕, 吉田 智章, 入江 清, 小柳 栄次: ”つくばチャレンジにおける屋外ナビゲーションシステムの問題分析と基本設計”, 日本ロボット学会誌, Vol.30, No.3, pp.262-270, 2012.
- [18] S.Thrun, W.Burgard and D.Rox, 上田 隆一 訳: ”確率ロボティクス”, 毎日コミュニケーションズ, 2007.
- [19] Paul J.Besl, Neil D.McKay, ”Method for Registration of 3-D Shapes.”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.14, No.2, pp.239-256, 1992.

- [20] Y.Chen, G.Medioni: "Object Modelling by Registration of Multiple Range Images", Proceedings of the 1991 IEEE International Conference on Robotics and Automation, Vol.10, No.3, pp.145-155, 1992.
- [21] N.Metropolis, S.Ulam, "The monte carlo method" Journal of the American statistical association Vol.44, No.247, pp 335-341, 1949.
- [22] A.F.M.Smith, A.E.Gelfand: "Bayesian statistics without tears: a sampling resampling perspective", The American Statistician, Vol.46, No.2, pp.84-88, 1992.
- [23] D.Fox, W.Burgard, S.Thrun: "Markov Localization for Mobile Robots in Dynamic Environments", Journal of Artificial Intelligence Research, Vol.11, pp.391-427, 1999.
- [24] 友納 正裕: "移動ロボットののための確率的な自己位置推定と地図構築", 日本ロボット学会誌, vol.29, No.5, pp.423-426, 2011.
- [25] F.Dellaert, W.Burgard, D.Fox, S.Thrun: "Using the CONDENSATION Algorithm for Robust, Vision-based Mobile Robot Localization", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.588-594, 1999.
- [26] 荒木 天外, 竹村 憲太郎, 怡土 順一, 松本 吉央, 高松 淳, 小笠原 司: "汎用三次元環境地図を用いた移動ロボットナビゲーションのための地図作成", 日本ロボット学会誌, Vol.28, No.1, pp.106-111, 2010.
- [27] 小森 谷清: "移動ロボットの位置決め技術", 精密工学会誌, Vol.65, No.10, pp.1402-1406, 1999.
- [28] 渡邊 賢, 長尾 確: "小型無人移動体との連携による個人用知的移動体の安全自動走行." 情報処理学会論文誌, Vol.53, No.11, pp.2599-2611, 2012.

- [29] S.Olufs, V.Markus: "An efficient area-based observation model for monte-carlo robot localization", Proceedings of International Conference on Intelligent Robots and Systems, pp.13-20, 2009.
- [30] E.Takeuchi, O.Kazunori, T.Satoshi: "Robust localization method based on free-space observation model using 3D-map", Proceedings of International Conference on Robotics and Biomimetics, pp.973-979, 2010.
- [31] 城殿 清澄, 三浦 純, 白井 良明: "誘導による移動経験に基づく視覚移動ロボットの自律走行", 日本ロボット学会誌, Vol.19, No.8, pp.1003-1009, 2001.
- [32] 川西 亮輔, 山下 淳, 金子 透: "全方位カメラを用いた周囲環境計測のための特徴点フローモデルによる位置姿勢推定", 映像情報メディア学会技術報告, Vol.33, No.11, pp.65-68, 2009.
- [33] M.Tomono: "3D Localization Based on Visual Odometry and Landmark Recognition Using Image Edge Points", Proceedings of International Conference on Intelligent Robots and Systems, pp.5953-5959, 2010.
- [34] M.Agrawal, K.Konolige: "Real-time localization in outdoor environments using stereo vision and inexpensive gps", International Conference on Pattern Recognition, Vol.3, pp.1063-1068, 2006.
- [35] 出村 公成, 中川 祐: "全方位カメラを用いたテンプレートマッチングに基づくモンテカルロ位置同定法", 日本ロボット学会誌, Vol.27, No.2, pp.249-257, 2009.
- [36] 江口 純司, 尾崎 功一: "DGPS を用いて設定したウェイポイント経路追従による実環境自律移動法の開発 -つくばチャレンジ 2010 における完走およびリタイヤに関する考察-", 日本ロボット学会誌, Vol.30, No.3 pp.314-323, 2012.

- [37] 北村 光教, 安岡 洋一, 間野 直哉, 鈴木 太郎: "高精度 GNSS 測位技術を用いたつくばチャレンジ 2010 における屋外自律移動システムの開発", 計測自動制御学会システムインテグレーション部門講演会 SI2010, 1A2-5, 2010.
- [38] 山崎 将史, 竹内 栄次郎, 大野 和則, 田所 諭: "三次元地形情報および GPS を用いたパーティクルフィルタによるマルチパスを考慮した自己位置推定", 日本ロボット学会誌, Vol.29, No.8, pp.702-709, 2011.
- [39] J.Borenstein, L.Feng: "UMBmark: A Benchmark Test for Measuring Odometry Errors in Mobile Robots", 1995 SPIE Conference on Mobile Robots, PA, pp.22-26, 1995.
- [40] J.Borenstein, L.Feng: "Measurement and correction of systematic odometry errors in mobile robots", IEEE Transactions on Robotics and Automation, pp.869-880, 1996.
- [41] 原 祥亮, 大島 章, 小野 幸彦, 網野 梓, 山本 健次郎: "人込み歩道環境に適応した自律移動技術の開発と実験機 Sofara-T を用いた実環境での評価", 日本ロボット学会誌, Vol.30, No.3, pp.287-295, 2012.
- [42] J.E. Dennis, R. B. Schnabel: "Numerical Methods for Unconstrained Optimization and Nonlinear Equations", SIAM Classics in Applied Mathematics, 1996.
- [43] A.Elfe: "Sonar-based real-world mapping and navigation", Transactions on Robotics and Automation, pp.249-265, 1987.
- [44] H.P.Moravec: "Sensor fusion in certainty grids for mobile robots." AI Magazine, Vol.9, No.2, pp.61-74, 1988.
- [45] J.Minguez, L.Montesano, F.Lamiroux: "Metric-Based Iterative Closest Point Scan Matching for Sensor Displacement Estimation", IEEE Transactions on Robotics, Vol.22, No.5, pp 1047-1054, 2006.
- [46] C.Andrea: "An ICP variant using a point-to-line metric", Proceedings of IEEE International Conference on Robotics and Automation, pp.19-25, 2008.

- [47] P.Biber, W.StraBer: "The Normal Distributions Transform: A New Approach to Laser Scan Matching", Proc. of IROS2003, pp.2743-2748, 2003.
- [48] Naoki Akai, Satoshi Hoshino, Kazumichi Inoue, Koichi Ozaki: "Monte Carlo Localization using Magnetic Sensor and LIDAR for Real World Navigation", Proceedings of the 2013 IEEE/SICE International Symposium on System Integration, MP2-M.3, 2013